VLSI architecture with a configurable data processing path based on serial distributed arithmetic

I E Tarasov¹ and D S Potekhin¹

¹MIREA - Russian Technological University, 78 Vernadsky Avenue, Moscow, 119454, Russia

E-mail: tarasov_i@mirea.ru

Abstract. The article considers the VLSI architecture intended for streaming data processing using a configurable path based on sequential distributed arithmetic. The development of specialized VLSIs is advisable if there is a need for serial production of devices, and the domain of application of such VLSIs decreases as the capabilities of the CPU, GPU and FPGA expand. The alternative approach is to build configurable VLSIs that can efficiently perform computations in related applications. The solution under consideration is focused on performing stream operations, including FIR filters and other tasks based on the operation “multiplication and accumulation”, for example, neural networks and deep learning algorithms. The architecture provides for compliance with the GALS (Globally Asynchronous, Locally Synchronous) approach, which allows placing many independent computing nodes in the VLSI. The characteristics of VLSI were evaluated and the areas of preferred application of the considered architecture were considered.

1. Introduction
The development of digital microelectronics technologies forms a circle of urgent problems for the development of a new element base for high-performance VLSI. In addition to specialized VLSI, the construction of computer systems can be made on the basis of a CPU, GPU or FPGA, and each of these classes of microcircuits is oriented to the mass market and has a combination of advantages and disadvantages. At the same time, increasing the cost of VLSI design limits the range of tasks for which the development of a specialized element base is advisable.

2. Problem overview
When choosing a VLSI architecture, it is necessary to take into account a number of problems, based both on the features of technological processes and on the need to adapt the architecture to the class of tasks being solved.

1. The increase in the density of components on a chip means an increase in power consumption per unit area. Since power is limited by the thermophysical characteristics of materials, an obstacle arises in optimizing performance. The attempt to implement a highly specialized hardware solution leads to the placement of densely packed components with a high switching activity coefficient on the VLSI chip. This means that the crystal overheats above the maximum permissible values [1, 2].

2. The increase in the area of the VLSI chip complicates the tracing of a high-frequency clock tree. This problem should be solved at the architectural stage by dividing the project into independent clock domains, the data transfer between which is carried out by special resynchronization schemes (for
example, based on dual-port static memory). The trend towards an increase in the number of clock domains is manifested for modern projects of specialized VLSI. For example, Xilinx FPGAs, starting with technology standards of 20 nm, do not allow tracing of the same clock signal throughout the FPGA chip. Instead, it is necessary to limit synchronous circuits to the boundaries of the clock region allocated on the chip. Also an example is the high-performance VLSI Epiphany-V, which has 1024 processor cores. It is made by 16 nm technological process and has 1152 clock domains [3].

3. The implementation of strongly coupled structures (such as grid or mesh) can complicate the implementation of VLSI at the stage of topological design. These problems are not obvious in the early stages of design, such as architectural and RTL design.

Summarizing, the developed VLSI for high-performance computing should meet the following requirements:
- focus on applications that complement the CPU, GPU and FPGA;
- a configurable data processing path that has two goals: expanding the range of tasks and reducing the density of simultaneously working components by introducing functional redundancy into the circuit;
- reducing the connectivity of components, limiting the area of clock networks.

Currently, there are a number of VLSI systems in industry that implement multi-core processor architectures or conveyor-configured systems [4-10]. This allows concluding about possible benefits of this architecture.

3. Designing VLSI for High Performance Computing Systems

The architecture under consideration involves the implementation of a set of configurable pipelines. The organization of the pipeline is shown in figure 1.

![Figure 1. Configurable pipeline.](image)

The functional units shown are controlled by independent configuration memory units. This also makes it possible to implement processing algorithms in which the number of sequential operations is greater than the number of functional devices. To do this, we must reuse the same functional device to perform a series of operations. For this, an input multiplexer is used, which feeds either data from a previous functional device or from the same device. Options of the distribution of operations among functional devices and clock cycles are shown in figure 2.
Figure 2. Options for configuring the pipeline for streaming execution and multiple operations at each stage of the pipeline.

Since the number of operations required to implement the target algorithm may differ, it is impractical to rigidly fix this number by setting a constant conveyor length. To organize chains from conveyors, the circuit shown in figure 3.

Figure 3. Organization of interaction between pipelines.
Here we can see that the input to the pipeline can be either an external data source or the output of the previous pipeline. This allows implementing both multi-channel signal processing systems using one pipeline, and one pipelined computer that uses sequential data processing in all conveyor chains.

As a functional device, it is advisable to consider the component of distributed arithmetic. Distributed arithmetic (SDA, serial distributed arithmetic) is an approach based on the execution of the sequence of operations “multiplication step” instead of parallel calculation of partial products with their addition on the basis of the adder tree. The operation "step of multiplication" is determined by the expression:

\[ \text{sum} = \text{sum} + (a \ll n) \times b(i) \]

where \( n \) – number of bits by which the first operand is shifted;

\( i \) – index of the bit of the second operand by which one-bit multiplication is performed.

The operation of single-bit multiplication in practice is reduced to performing the LOGICAL AND operation, and the same bit of the second operand is supplied to the second input of all logic gates. Compared to the adder tree, stepwise multiplication is the opposite option, where instead of multiplying over 1 clock on N adders, the same operation is performed over N clocks on one adder.

In figure 4 shows the main components of a sequential multiplication scheme. One of the operands \((A)\) is shifted to the left by a predetermined number of bits using the barrel shifter. The second number is shifted to the same number of bits to the right, and its least significant bit is used as a factor. To obtain the result of multiplying N bits by a 1-bit binary number, it is sufficient to use the multi-bit element and, for which each of the bits of the result is determined by the expression: \( M(i) = A(i) \) and \( B(0) \).

In addition to the operation "step of multiplication with accumulation", for functional devices, we can consider other options for basic operations. The example would be the CORDIC algorithm for sequentially computing transcendental functions. In combination with nodes performing multiplication with accumulation, this allows realizing the operations of stream convolution with kernels generated in VLSI in real time.

**Figure 4.** Scheme for the sequential execution of multiplication by the implementation of the shift-addition.
When implementing functional devices with the ability to perform the “multiply with accumulation” step, VLSI performance depends on the capacity of the multiplied operands. For example, FPGAs use DSP hardware blocks having a bit width of operands of $18 \times 18$, $18 \times 25$, or $18 \times 27$. Using data of lower capacity does not increase the number of operations that can be performed using such blocks. However, using sequential distributed arithmetic, it becomes possible to use exactly the required number of blocks of sequential multiplication.

The VLSI architecture is shown in figure 5.

![Proposed VLSI architecture](image)

**Figure 5.** Proposed VLSI architecture.

VLSI consists of an input switch with $N$ inputs, $M$ processing channels and $K$ postprocessors. The amount of resources of these types can formally be different, which allows balancing the architecture in the design process, adapting it to tasks with different requirements for productivity and post-processing complexity. A wide range of existing solutions can be used as a postprocessor, such as ARM, MIPS, RISC-V, or specialized general-purpose processors.

### 4. Overview of results

An overview of the characteristics of VLSI, which can be implemented on the basis of the presented architecture, was carried out using FPGA CAD. The obtained characteristics were extrapolated to the VLSI hardware platform. The results of the comparative analysis are given in table 1.

**Table 1.** Results of a comparative analysis of the evaluated VLSI project and existing hardware platforms for digital signal processing.

| Parameter                        | Xilinx Zynq 7000 | Xilinx Zynq UltraScale+ | Xilinx Virtex UltraScale+ | VLSI evaluation |
|----------------------------------|------------------|------------------------|---------------------------|-----------------|
| Technology node, nm              | 28               | 16                     | 16                        | 28              |
| DSP blocks                       | 2020             | 2520                   | 12288                     | 4000            |
| Clock speed, MHz * for FIR filterIP-core | 350*           | 400*                   | 400*                      | 800 - 1000      |
| DSP GMAC/s                       | 700              | 1000                   | 4900                      | 3200-4000       |
It can be seen that the evaluation of the project involves the implementation of mainly digital signal processing, without components such as programmable FPGA cells. In addition, in VLSI there are no such components as MGT and memory blocks occupying the area in the FPGA, which explains the significantly better VLSI parameters compared to the components of programmable logic chips. One of targeted area is software-defined radio using high-order kernel functions [12].

5. Conclusion
The considered VLSI architecture is intended for joint use with such high-performance solutions as CPU, GPU and FPGA. The architecture is focused on pipelined computing, for example, digital signal processing, and can be used in pre-filtering streaming data in industrial, medical and scientific problems. As practical steps, VLSI modeling at the system level with the transition to a model based on the FPGA is supposed.

References
[1] Esmaeilzadeh H, Blem E, Amant R S, Sankaralingam K and Burger D 2011 Proc. of 38th Int. Sym. on Computer Architecture (San Jose, California, USA)
[2] Hennessy J L and Patterson D A 2017 Computer Architecture. 6th Edition. A Quantitative Approach (The Morgan Kaufmann Series in Computer Architecture and Design) p 936
[3] Olofsson A 2016 Epiphany-V: A 1024 processor 64-bit RISC System-On-Chip
https://www.parallella.org/wp-content/uploads/2016/10/e5_1024core_soc.pdf
[4] Japan's own processor "PEZY-SC2" and equipped with supercomputers "Gyoko" Details. https://pc.watch.impress.co.jp/docs/news/1091458.html
[5] Jouppi N P and Wall D W 1989 Proc. of 3rd Int. Conf. on Architectural Support for Programming Languages and Operating Systems (Boston, Massachusetts, USA)
[6] Lu W, Yan G, Li J, Gong S, Han Y and Li X 2017 Proc. of IEEE Int. Symp. on High Performance Computer Architecture (HPCA,2017) 553
[7] Wu S-Y et al 2013 Proc. of IEEE International Electron Devices Meeting (9-11 Dec. 2013, Washington, DC, USA)
[8] Jouppi N P, Young C, Patil N, Patterson D, Agrawal G, Bajwa R, Bates S, Bhatia S, Boden N, Borchers A and et al 2017 Proc. of 44th Annual Int. Symp. on Computer Architecture (ISCA 2017) p 1
[9] NVDLA deep learning accelerator 2017 http://nvdl.org/
[10] Kwon H, Samajdar A and Krishna T 2018 Proc. of Twenty-Third Int. Conf. on Architectural Support for Programming Languages and Operating Systems (ACM SIGPLAN Notices) vol 53(2) p 461
[11] http://www.xilinx.com/
[12] Tarasov I E and Potekhin D S 2018 RUSSIAN TECHNOLOGICAL JOURNAL vol 6 pp 41-54. DOI: 10.32362/2500-316X-2018-6-6-41-54