# Speed-Independent Fused Multiply Add and Subtract Unit*

Yuri Stepchenkov, Victor Zakharov, Yuri Rogdestvenski, Yuri Diachenko,
Nickolaj Morozov and Dmitri Stepchenkov
*Institute of Informatics Problems, Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences (IPI FRC CSC RAS), IPI RAS, Moscow, Russian Federation*
*{YStepchenkov, VZakharov, YRogdest, YDiachenko, NMorozov, DStepchenkov}@ipiran.ru*

## Abstract

*Speed-independent fused multiply-add-subtract unit is offered together with test environment providing full verification of its performance and workability in all range of the environment conditions. It complies with IEEE 754 Standard, and performs double and single precision operations at three operands. The unit is implemented as a two-channel with a common input and output. Each channel is a pipeline with four stages. Multiplier is implemented on the modified Booth algorithm using self-timed redundant code. The unit was designed on a base of standard CMOS process with 65 nm design rules and has 3.15 Gigaflops performance and less than 2 ns latency.*

## 1. Introduction

Fused Multiply-Add (FMA) operation is a standard operation of modern computers. Adding an operation of subtracting third operand from a product of first two operands extends the functionality of the device. Hereinafter we will refer to it as FMAS (Fused Multiply-Add & Subtract) unit.

A lot of publications presented implementations of synchronous FMA unit (for example, [1, 2]) are known. Approaches to asynchronous FMA implementation are investigated to a much lesser extent [3, 4], and are virtually non-existent implementations of really self-timed (ST) units of this type, which due to their characteristics would match the Speed-Independent (SI) circuits, the proper function of which is not dependent on the delays in elements of the circuit.

SI circuits reduce energy consumption due to non-use of clock generator and "clock tree" and natural switching into energy-saving mode of the equipment part not used in the current cycle of information processing. In addition, the SI circuits keep their workability at ultra-low values of supply voltages. This opens up broad prospects for designing energy-efficient products. The tradeoff for such benefits is hardware redundancy and overhead delays for indication and spacer phase in the SI work. However, proper design of SI circuits can substantially reduce this redundancy, and in some classes of computing devices [5] obtain results even better than in synchronous circuits.

This article presents the results of designing 64-bit SI-FMAS unit of gigaflops range complying with IEEE 754 Standard. The purpose of the investigation was to develop speed-independent FMAS unit demonstrating maximum performance, low latency and testability, which provides measuring unit's features including its workability range. Supposed operation of the described unit with synchronous environment requires using special means of its testing for demonstrating features of the true SI units interfacing with self-timed environment.

## 2. SI-FMAS Implementation

Described SI-FMAS unit is developed on the basis of the SI floating-point coprocessor (SI-FPC) unit [6]. Comparing to it, the offered device differs in two times higher performance, lower latency and testing capabilities in a wide range of supply voltage and ambient temperature. It was designed to operate with a synchronous environment.

### 2.1. Block diagram

The scope of SI-FMAS as basic computing element of a distributed computing network puts forward minimum energy consumption in high enough performance as the primary requirement. This is accomplished using relatively low frequency and large number of SI-FMAS nodes composing one large computing system.

Two phases in the work of any SI circuits (active – work, and pause – spacer) suggests the idea of using

* The reported study was partially funded by RAS Research Program (project 0063-2015-0015 RAS 1.33P) in IPI FRC CSC RAS.

978-1-5090-0693-9/16/$31.00 ©2016 IEEE

two parallel channels with alternated phase of work: switch to the work phase of first channel is accompanied by switching second channel into spacer phase, and vice versa. Such approach greatly reduces a wasteful waiting time of the synchronous environment, and almost doubles the performance of SI-FMAS.

Figure 1 presents a block diagram of the SI-FMAS implementing this idea. Information inputs are the operands (*X*, *Y*, *Z*), attributes of operation (*R*), initial reset (*Rst*), clock (*Clk*) signal and data processing permission (Start). The results of the operation are sum (*FMA*), difference (*FMS*) and the appropriate operation flags (*RFA*, *RFS*).

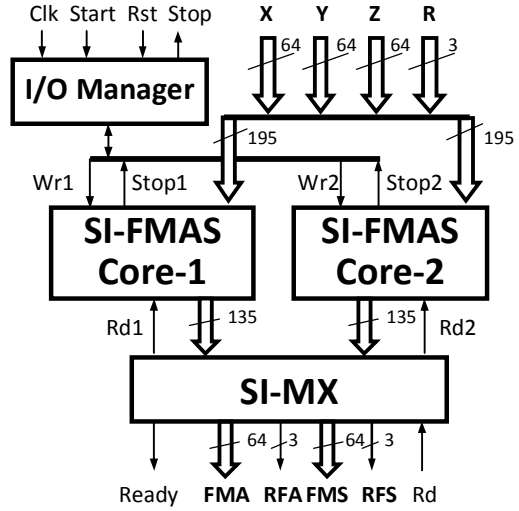


**Figure 1. Block diagram of SI-FMAS**

Output *Stop* prevents the possible loss of input data due to the unwillingness of the SI-FMAS to take a new three operands for processing. The *Ready* output signals synchronous environment about the successful completion of regular operation. Input *Rd* confirms successful reading result by synchronous environment.

SI-FPC described in [6] and modified for work with synchronous environment is utilized as SI-FMAS core. Figure 2 shows block diagram of the SI-FMAS core.

SI-FMAS core contains input and output FIFO. They improve the performance of the SI-FMAS when working with synchronous environment [7] due to the buffering of the data stream. Input and output FIFO are implemented as semi-dense SI shift registers [8, fig. 11.9] with 3 words capacity. They form the first and last stages of SI-FMAS core pipeline respectively. New operand arriving such FIFO automatically moves to free cell, the nearest to the output. When full, the input FIFO produces *Stop* signal, forcing synchronous environment pause to "pump" SI-FMAS unit by input operands.

Additional feature of the semi-dense SI FIFO is that information input is a single-rail signal rather than dual-rail or bi-phase signal that are traditional for self-timed circuits. This helps one to build an interface between SI-FMAS and synchronous environment, reducing the number of interconnection wires and simplifying layout realization of the unit.

Output SI FIFO generates *Ready* signal confirming the readiness of the regular operation results. It keeps current operation result until the synchronous environment forms *Rd* signal as an acknowledge of successful reading operation result from SI-FMAS unit.

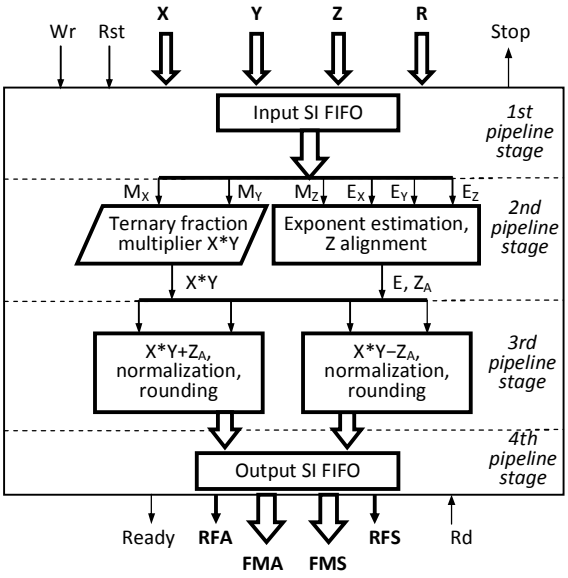


**Figure 2. Block diagram of SI-FMAS Core**

Logically SI FIFO corresponds to the semi-dense register [8] implementation, but takes into account the circuitry restrictions of the industrial standard cell libraries and provides initial reset of the FIFO by *Rst* signal.

Publications [6, 7] describe in detail the rest part of the SI-FMAS core implementation.

## 2.2. SI-FMAS Indication

One reason of the large complexity of the SI circuit implementations is the obligatory presence of a subcircuit indicating all circuit transitions in each phase of its work, which ensures correct operation of any SI circuit. The indication subcircuit is the "bottleneck" of any SI circuit, especially multi-bit unit, dramatically slowing its work.

Problem of an optimization of multi-bit SI unit indication has already been discussed in [6]. The basis for a solution of this problem is dual-rail and self-timed redundant coding used in SI-FMAS unit that guarantees no more than single switch of all elements of a combinational part of the pipeline during its transferring from spacer to work phase. Therefore, the appear-

ance of work condition after spacer on all information outputs of the combinational part of the SI circuit ensures the result readiness.

Proposed in [6] solution of the multi-bit indication problem has allowed for significant reducing complexity of the indication subcircuit and improving its performance. However, a more detailed study of the problem for different classes of computing units has discovered that strongly simplified indication subcircuit loses a property of constant fault detection (cell's output "sticking" in one state).

In this regard, the proposed SI-FMAS has an indication of combinational circuits simplified only at its first stage. Depending on the type of spacer (zero or unit) and the library of standard elements, this allows for reducing number of the outputs of first indication subcircuit stage by factor of 1.5 – 2 and respective simplifying of its second and subsequent stages.

## 3. Test environment

One of the most tedious tasks in designing modern digital units is the workability determination in the range of supply voltages and ambient temperatures. On the one hand, using request-acknowledge interaction between SI units makes easier the solution of this problem, because SI device itself generates signals indicating its ability to accept new input data and the readiness of the result. On the other hand, the purpose of this study is to determine the characteristics of SI unit itself. Therefore, one should exclude a negative impact of the synchronous environment because of its possible unstable behavior at variations in the supply voltage and ambient temperature.

Figure 3 shows a block diagram of the test environment for SI-FMAS. Method of determining workability range of SI-FMAS is the following. A set of input operands is fixed. Calculations in SI-FMAS are started at nominal supply voltage and normal ambient temperature. The first result is written to the reference register. Further write to the reference register is locked, until next reset signal appears. Then SI-FMAS is switched into self-timed (cyclic) mode of operation. In this mode, an output signal *OK* generated by the comparator after successful comparing current operation result with the result kept in the reference register permits starting next operation.

In this mode, synchronous clock *Clk* and permission signal *Start* are not used. Thus, all subsequent results of processing fixed operand set are compared with the content of the reference register. Regular appearance of high-level signal at the output *OK* says about keeping workability by SI-FMAS unit.

Changing supply voltage and ambient temperature and monitoring *OK* output (e.g. by an oscilloscope) during this test allows one to determine the workability range of the SI-FMAS unit.

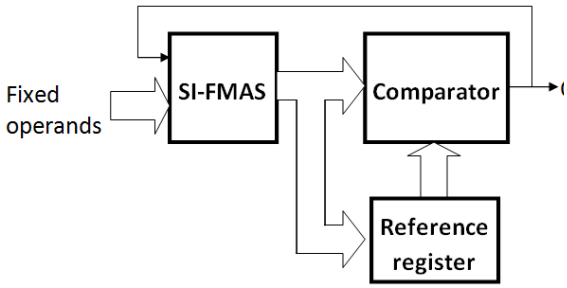


**Figure 3. Test environment of SI-FMAS**

Note that the adjustable supply voltage is served to SI-FMAS separately from the synchronous environment. Supply voltage of the latter is maintained at nominal level in order to avoid the influence of the synchronous environment on the signal *OK* generation. To make *OK* output observable at any voltage supplied to SI-FMAS (especially for values less than nominal supply voltage of the synchronous environment) one should utilize level converter for this signal before outputting it to a pad.

Modeling circuit of the SI-FMAS unit extracted from layout designed for 65 nm CMOS process by means of Ultrasim (Cadence), has confirmed its successful workability at temperatures from $-63^0$C to $125^0$C and supply voltages from 0.3V to 2.0V.

## 4. SI-FMAS parameters

SI-FMAS was designed for standard 65 nm CMOS process with 6 metal layers. Parameters for SI-FMAS are given in the Table for operation mode with synchronous environment in comparison with synchronous analogue having nearest clock frequency [9].

**Table. SI-FMAS's parameters**

| Parameter | Analog | SI-FMAS |
|---|---|---|
| Die size, $mm^2$ | 0.312 | 1.04 |
| Performance, Gflops | 2.06 | 3.15 |
| Latency, ns | 10.8 | 1.84 |
| Die size efficiency, $mm^2$/Gflops | 0.151 | 0.298 |
| Work range on power supply $V_{DD}$ | $V_{DD} \pm 10\%$ | $V_{th}...V_{BD}$ |

Time and energy parameters are based on simulation with parasitic capacitances and resistors extracted from layout for statistically reliable set of input operand combinations for double and single-precision.

Note that SI-FMAS has greater functionality than its synchronous analogue: each cycle it processes either three double precision operands, or two single-precision operand triplets, calculating at the same time both sum and difference between the product of first two operands and third operand.

In addition, it has a much wider range of performance, limited only by the threshold voltage of the CMOS transistors ($V_{th}$) and breakdown voltage of semiconductor structures ($V_{BD}$), and stops at detecting constant fault [8]. Greater complexity and, therefore, more power consumption is a penalty of these benefits. Energy consumption can be reduced to the desired value by decreasing supply voltage, accompanied by corresponding performance decrease. Due to fewer pipeline stages, SI-FMAS has latency by factor of 5.9 less than for synchronous analogue.

Checking self-timed features of SI-FMAS using event analysis program ASPECT [10] has proved its affiliation to SI unit class.

Proposed SI-FMAS implementation allows for sufficiently easy increasing number of concurrent SI-FMAS units composing one processor of the super-computers. This contributes to the solution of problems of improving their performance and reliability, such as doubling SI-FMAS units, which are fully self-checked relatively constant faults, to build their failsafe variants.

## 5. Conclusions

Applying SI-circuitry for implementing modern computing systems allows one to effectively use hardware for controlling reliability and validity of the calculation results.

Scientific novelty of the investigation consists in designing speed independent pipe-lined 64-bit FMAS unit demonstrating high average performance and low latency, as well as in developing test tools proving that suggested unit is true SI unit even in synchronous environment. SI-FMAS with two concurrent blocks of FMAS designed for 65 nm CMOS process has the average performance of 3.15 Gflops at typical conditions and latency less than 2 ns.

Usage of the redundant self-timed coding, simplified indication and the minimum number of pipeline stages ensure the development of competitive on performance 64-bit unit implementing FMAS operation and having all the advantages of SI devices: full self-checking regarding constant faults, saving workability at ultra-low supply voltages.

To reduce the hardware costs and energy consumption of the SI-FMAS, we are going to study an implementation of its multiplier as a two-stage pipeline calculating two halves of the Wallace tree for most and least significant bits separately and consistently. This should remain the same performance of the SI-FMAS while using one computing channel instead of two identical channels due to co-processing dual-rail calculation discipline in one channel at two pipeline stages.

## 6. References


[1] R.V.K. Pillai, S.Y.A. Shah, A.J. Al-Khalili, and D. Al-Khalili, "Low power floating point MAFs – A comparative study", *Sixth International Symposium on Signal Processing and its Applications*, Kuala Lumpur, 2001, V. 1, pp. 284-287.

[2] P.-M. Seidel, "Multiple path IEEE floating-point Fused Multiply-Add", *46th IEEE International Midwest Symposium on Circuits and Systems*, Cairo, Egypt, 2003, pp. 1359–1362.

[3] J.R. Noche, and J.C. Araneta, "An asynchronous IEEE floating-point arithmetic unit", *Science Diliman*, Philippines, 2007, V.19, No.2, pp. 12–22.

[4] R. Manohar, and B.R. Sheikh, "Operand-optimized asynchronous floating-point units and method of use therefore", *US patent*, № 20130124592. May 2013.

[5] Y. Stepchenkov, Y. Diachenko, V. Zakharov, Y. Rogdestvenski, N. Morozov, and D. Stepchenkov, "Quasi-delay-insensitive computing device: methodological aspects and practical implementation", *International Workshop on power and timing modeling, optimization and simulation (PATMOS'2009)*, Delft, Netherlands, 2009, pp. 276–285.

[6] Y.A. Stepchenkov, V.N. Zakharov, Y.V. Rogdestvenski, Y.G. Diachenko, N.V. Morozov, and D.Y. Stepchenkov, "Speed-Independent Floating Point Coprocessor", *IEEE East-West Design and Test Symposium*, Batumi, Georgia, September 26-29, 2015, pp. 111 - 114.

[7] I.A. Sokolov, Y.A. Stepchenkov, Y.V. Rogdestvenski, and Y.G. Diachenko, "Self-timed multiply-add unit of gigaflops range: methodological aspects", *Problems of advanced micro- and nano-electronics systems development*, Moscow, IPPM RAS, 2014, V. IV, pp. 51-56 (In Russian).

[8] *Self-Timed Control of Concurrent Processes: The Design of Aperiodic Logical Circuits in Computers and Discrete Systems* (V.I. Varshavsky, ed.), Kluwer Academic Publishers, 1990, 408 p.

[9] S. Galal, and M. Horowitz, "Energy-Efficient Floating-Point Unit Design", *IEEE Transactions on computers*, 2011, V. 60, No.7, pp. 913–922.

[10] Y. V. Rogdestvenski, N. V. Morozov, and A. Rozhdestvenskene, " ASPECT: A subsystem for event analysis of self-timed circuits", *Problems of advanced micro- and nano-electronics systems development*, Moscow, IPPM RAS, 2010, pp. 26–31 (In Russian).