

**ОЦЕНКА ИСПОЛЬЗОВАНИЯ ПАМЯТИ  
ПРОГРАММИРУЕМЫХ ЛОГИЧЕСКИХ ИНТЕГРАЛЬНЫХ СХЕМ  
ПРОСТЫМИ АЛГОРИТМАМИ ТИПА БЫСТРОГО  
ПРЕОБРАЗОВАНИЯ ФУРЬЕ С ПОМОЩЬЮ МОДУЛЯ LabVIEW FPGA**

**Е.М. Череданова**

pankooova@mail.ru  
SPIN-код: 1619-5499

**Е.А. Мамченко**

liza.98.98@mail.ru  
SPIN-код: 2887-3715

**МГТУ им. Н.Э. Баумана, Москва, Российская Федерация**

---

**Аннотация**

*Целью работы является исследование загруженности программируемых логических интегральных схем (ПЛИС) типовыми алгоритмами для дальнейшей оптимизации системы за счет перекладывания части функций на контроллер. В этих целях реализован алгоритм быстрого преобразования Фурье (БПФ) на ПЛИС в среде прикладного программного обеспечения LabVIEW с использованием программного модуля LabVIEW FPGA. Созданное приложение состоит из трех модулей: отдельного виртуального прибора, выполненного на ПЛИС (FPGA VI), виртуального прибора на контроллере реального времени (Real Time VI) и виртуального прибора на персональном компьютере с операционной системой Windows (Host VI). В результате исследования осуществлена оценка эффективности использования памяти на ПЛИС с помощью анализа отчета о результатах компиляции бинарного файла при замене встроенного программного обеспечения ПЛИС через модуль LabVIEW FPGA, что позволяет оценить объем памяти ПЛИС, занимаемой простейшими алгоритмами типа БПФ.*

**Ключевые слова**

*Быстрое преобразование Фурье, программируемые логические интегральные схемы, FPGA, LabVIEW, виртуальный прибор, бинарный файл, ПЛИС*

Поступила в редакцию 06.11.2018  
© МГТУ им. Н.Э. Баумана, 2018

Задачей данного исследования является реализация алгоритма быстрого преобразования Фурье (БПФ) на программируемых логических интегральных схемах (ПЛИС) в программном модуле LabVIEW FPGA и оценка объема занимаемой на ПЛИС памяти. Это позволит понять, как много ресурсов ПЛИС занимают типовые простые алгоритмы типа БПФ, а в дальнейшем — оценить целесообразность оптимизации аппаратно-программных систем за счет перекладывания часть вычислительных функций на контроллер и ПЛИС [1].

Сформируем испытательный стенд для проведения исследования, включающий в себя набор аппаратно-программных средств и специального программного обеспечения, устанавливаемого на персональный компьютер (ПК).

К аппаратно-программным средствам относятся:

- источник питания NI PS-15;
- шасси с интегрированным контроллером реального времени NI cRIO-9075;
- четырехканальный модуль для регистрации акустических и вибросигналов NI cRIO-9234;
- сетевой кабель длиной 0,5 м класса UTP (без экранирования).



**Рис. 1.** Схема подключения аппаратной части исследовательского стенда

Источник питания NI PS-15, подключенный к электрической сети переменного тока общего назначения (220 В, 50 Гц), на выходе выдает 24 В постоянного тока силой 5 А, необходимого для питания шасси NI cRIO-9075. Модуль регистрации акустических и вибросигналов NI cRIO-9234 подключен во второй слот для модулей ввода/вывода С-серии (через последовательный порт RS-232) шасси NI cRIO-9075. Само шасси подключено к ПК с помощью сетевого кабеля. Схема подключения аппаратной части исследовательского стенда представлена на рис. 1.

На ПК установлен следующий комплект специального программного обеспечения [2]:

- NI LabVIEW 2015 (15.0f, 32-битная версия среды разработки приложений);
- NI CompactRIO Device Drivers May 2017 (сборник драйверов устройств NI CompactRIO);
- LabVIEW 2015 FPGA Module (модуль программируемых пользователем вентильных матриц);
- LabVIEW 2015 Real-Time Module (модуль реального времени);
- LabVIEW 2015 FPGA Module Xilinx Tools 14.7 (локальный компилятор для прошивок устройств Xilinx).

Разрабатываемое для CompactRIO приложение будет использовать три отдельных процессора: на компьютере с операционной системой (ОС) Windows, на контроллере реального времени и ядро ПЛИС. Отметим, что на контроллере CompactRIO вместо Windows установлена ОС реального времени, а на ПЛИС вообще нет ОС, код реализуется непосредственно на аппаратном уровне [3].

Таким образом, файл проекта разрабатываемого приложения будет включать три модуля (рис. 2):

- отдельный виртуальный прибор, выполненный на ПЛИС (FPGA VI);
- виртуальный прибор на контроллере реального времени (Real Time VI);
- виртуальный прибор на ПК с ОС Windows (Host VI).

Виртуальный прибор, выполненный на ПЛИС, реализован с минимально необходимыми функциями и включает в себя элементы, демонстрирующие значения выходного напряжения модуля NI cRIO-9234, а также кнопки начала

и окончания сеанса получения данных от модуля. Управление модулем осуществляется через элемент FPGA I/O Node. Лицевая панель и блок-диаграмма виртуального прибора, выполненного ПЛИС, представлены на рис. 3.

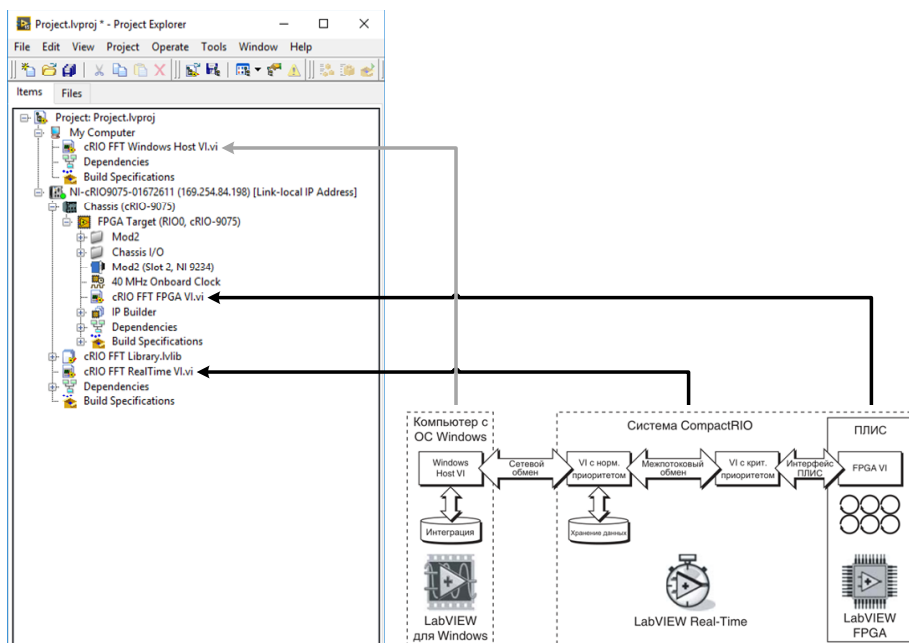
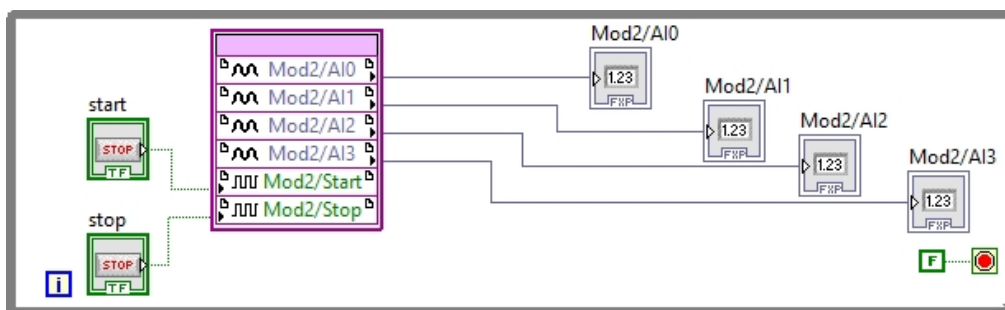


Рис. 2. Архитектура создаваемого приложения



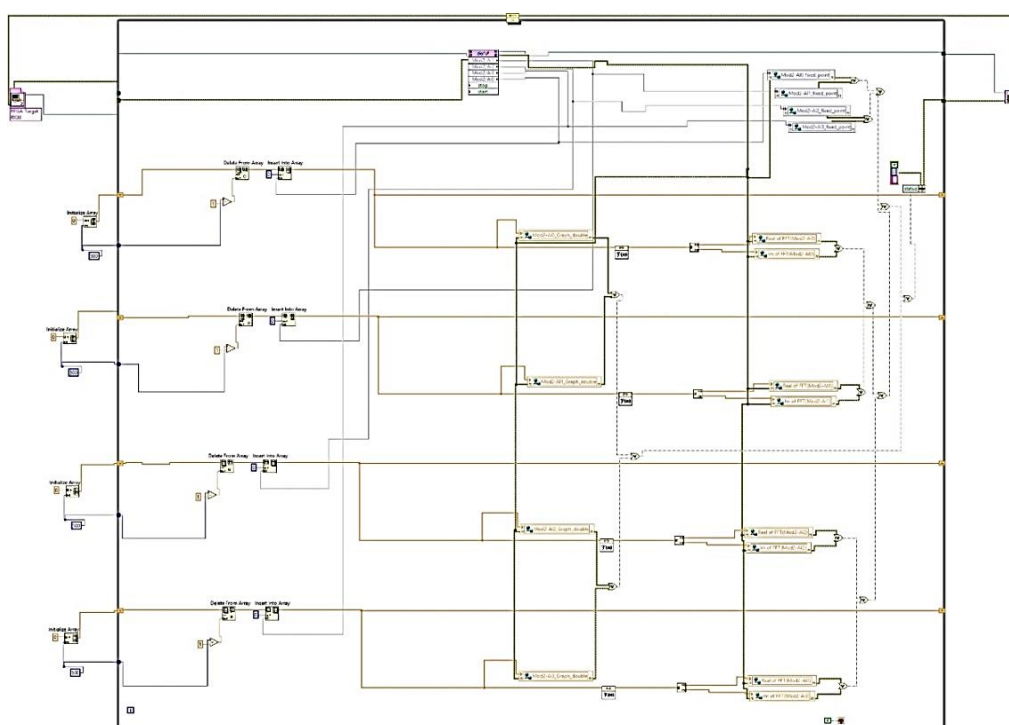
Рис. 3. Лицевая панель и блок-диаграмма виртуального прибора, выполненного на ПЛИС



Построение виртуального прибора на контроллере реального времени имеет ряд особенностей. Во-первых, лицевая панель прибора отсутствует за нена-

добностью, так как Real Time VI является посредником, передающим данные между виртуальными приборами, выполняющимися на ПЛИС и в ОС Windows. Во-вторых, значения выходного напряжения, а также спектральные данные, в том числе отдельно мнимая и действительная части (с четырех входов модуля NI cRIO-9234) для удобства сведены в переменные, хранящиеся в отдельной библиотеке cRIO FFT Library.lvlib проекта [4]. Это позволяет упростить доступ к вышеуказанной информации из виртуального прибора, выполняемого на ПК с ОС Windows.

Блок-диаграмма виртуального прибора на контроллере реального времени представлена на рис. 4.



**Рис. 4.** Блок-диаграмма виртуального прибора на контроллере реального времени

Виртуальный прибор, выполненный на ПК с ОС Windows, для отбора необходимых значений обращается к библиотеке проекта. Лицевая панель приложения демонстрирует пользователю значения спектра сигналов, отображаемые на графиках, данные о мнимой и действительной части и текущие значения выходного напряжения четырехканального модуля NI cRIO-9234. Кроме того, в Host VI предусмотрена возможность введения задержки обновления и отображения указанных выше значений благодаря использованию таймера цикла Loop Timer [5].

Лицевая панель и блок-диаграмма виртуального прибора, выполненного на ПК с ОС Windows, представлены на рис. 5 и 6 соответственно.

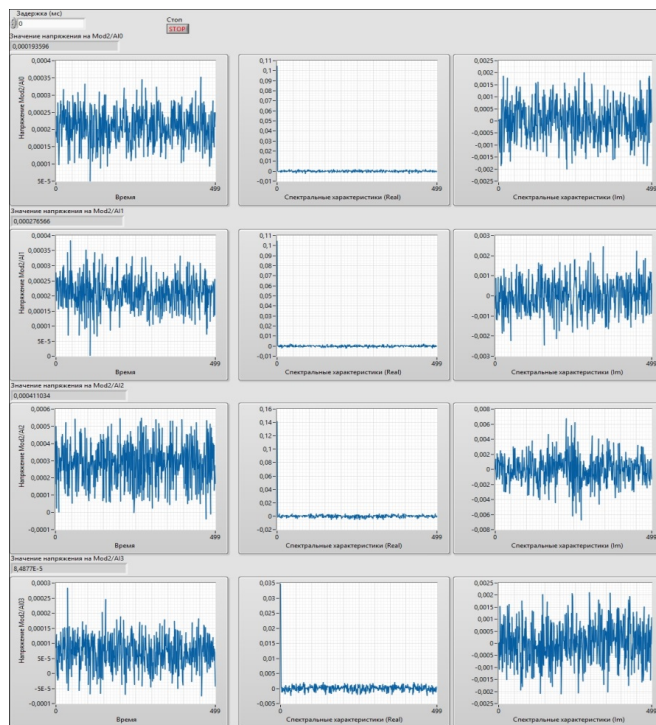


Рис. 5. Лицевая панель виртуального прибора, выполненного на ПК с ОС Windows

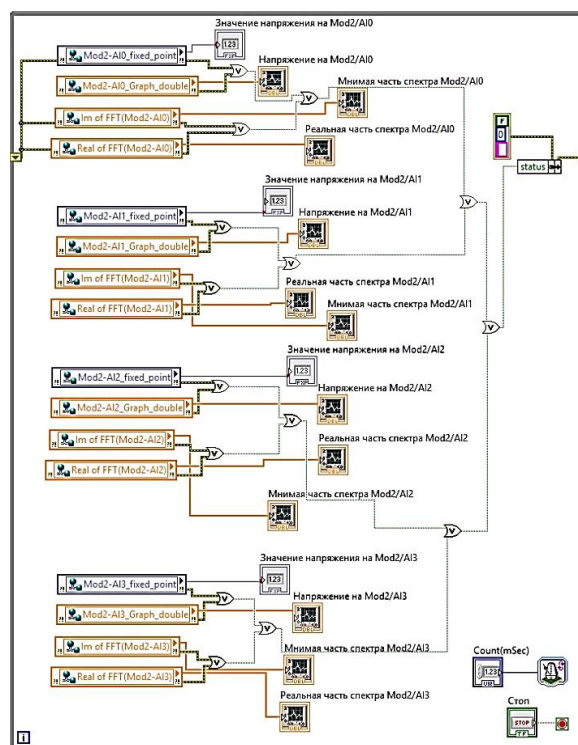


Рис. 6. Блок-диаграмма виртуального прибора, выполненного на ПК с ОС Windows

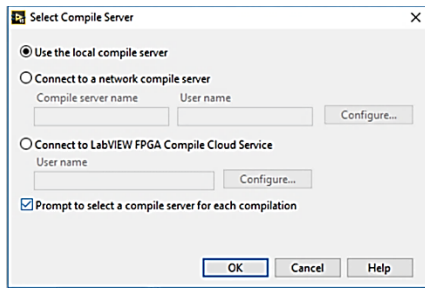


Рис. 7. Выбор параметров компиляции FPGA VI

После завершения построения трех виртуальных приборов проекта необходимо скомпилировать виртуальный прибор FPGA VI, загрузить его на ПЛИС и оценить объем используемой прибором памяти схемы [6].

Для этого запустим FPGA VI, инициализировав процесс компиляции сервером (рис. 7).

Компиляция файлов FPGA VI проекта занимает несколько минут. В это

время в отдельном окне (рис. 8) отображается статус выполнения различных этапов компиляции, а также путь к рабочей папке [7].

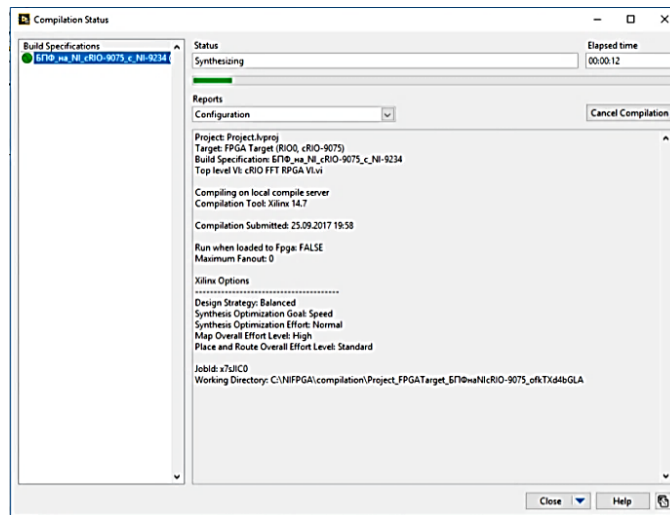


Рис. 8. Окно статуса компиляции FPGA VI

В результате компиляции в папке проекта создается папка FPGA Bitfiles, в которой находится скомпилированный бинарный файл (bitstream file) с расширением .lvbitx, полностью определяющий конфигурацию ПЛИС. При запуске FPGA VI бинарный файл загружается на ПЛИС, т. е. происходит процесс ее конфигурирования.

Кроме того, бинарный файл можно записать во встроенную флеш-память устройства NI cRIO-9234, что позволит автоматически загружать FPGA VI на ПЛИС при включении питания модуля. Для этого в окне проекта необходимо вызвать контекстное меню объекта FPGA Target, щелкнув по нему правой клавишей мыши, и выбрать пункт RIO Device Setup [8]. В появившемся окне следует щелкнуть по значку папки и, выбрав скомпилированный бинарный файл, нажать кнопку Download Bitfile, после чего программа оповестит об успешном завершении операции (рис. 9).

Оценка использования памяти на ПЛИС заключается в анализе файла отчета о результатах проведения компиляции бинарного файла при запуске FPGA VI. Вышеуказанный файл находится в рабочей папке компилятора и называется lvXilinxLog.txt. Информация, содержащаяся в подразделе Writing output files... файла lvXilinxLog.txt, представлена на рис. 10.

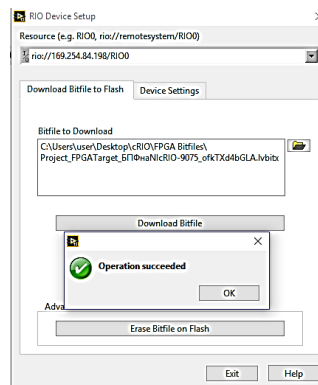


Рис. 9. Запись бинарного файла во встроенную флеш-память устройства NI cRIO-9234

Design Summary:

Number of errors: 0

Number of warnings: 240

Slice Logic Utilization:

<b>Number of Slice Registers:</b>	<b>1,592 out of 30,064</b>	<b>5%</b>
Number used as Flip Flops:	1,526	
Number used as Latches:	66	
Number used as Latch-thrus:	0	
Number used as AND/OR logics:	0	
<b>Number of Slice LUTs:</b>	<b>1,424 out of 15,032</b>	<b>9%</b>
Number used as logic:	1,329 out of 15,032	8%
Number using O6 output only:	1,035	
Number using O5 output only:	6	
Number using O5 and O6:	288	
Number used as ROM:	0	
Number used as Memory:	19 out of 3,664	1%
Number used as Dual Port RAM:	0	
Number used as Single Port RAM:	18	
Number using O6 output only:	7	
Number using O5 output only:	1	
Number using O5 and O6:	10	
Number used as Shift Register:	1	
Number using O6 output only:	1	
Number using O5 output only:	0	
Number using O5 and O6:	0	
Number used exclusively as route-thrus:	76	
Number with same-slice register load:	76	
Number with same-slice carry load:	0	
Number with other load:	0	

Рис. 10. Отчет о результатах компиляции бинарного файла при запуске FPGA VI

В данном фрагменте файла-отчета указано количество задействованных регистров секций ПЛИС (в графе Number of slice registers) и ее логических таблиц (Number of slice LUTs, LUTs — lookup tables). В общем случае количество занимаемой скомпилированным кодом виртуального прибора памяти ПЛИС равно процентному отношению задействованных регистров секций (1592) к свободным (30064) и составляет 5,3 % [9]. Однако в рамках данной работы можно вывести интегральный коэффициент задействования памяти ПЛИС, учитывающий не только использование регистров секций, но и логических таблиц. Значение интегрального коэффициента использования памяти ПЛИС будет равно процентному соотношению суммы задействованных регистров секций и логических таблиц ПЛИС к их общему доступному количеству. Расчет значения интегрального показателя представлен в таблице.

#### Расчет значения интегрального показателя задействования памяти ПЛИС

Наименование параметра	Значения параметра, ед.		Процентное соотношение задействованных элементов к общему числу доступных, %
	Задействовано	Всего доступно	
Регистры секций ПЛИС	1 592	30 064	5,3
Логические таблицы ПЛИС	1 424	15 032	9,5
Интегральный коэффициент использования памяти ПЛИС	3 016	45 096	6,9

Из таблицы видно, что объем занимаемой памяти на ПЛИС составляет около 70 %. Целесообразно оптимизировать аппаратно-программную систему за счет перекалывания части вычислительных функций на контроллер и ПЛИС. Данная оптимизация приведет к улучшению быстродействия системы, что в наше время очень актуально.

#### Литература

- [1] Турыгин И.Г. *Метод выбора программируемых логических интегральных схем на основе целевого функционала при проектировании устройств цифровой обработки информации*. Дисс. ... канд. техн. наук. Пенза, 2014, 134 с.
- [2] Галанина Н.А. *Методы и вычислительные устройства цифровой обработки сигналов в системе остаточных классов*. Автореф. дисс. ... док. техн. наук. Казань, 2010, 32 с.
- [3] Авяшкиева Н.С. Алгоритм быстрого преобразования Фурье в системе остаточных классов. *Технические науки - от теории к практике. Сб. ст. по матер. V межд. науч.-практ. конф.* Новосибирск, СибАК, 2011, с. 79–83.
- [4] Батоврин В.К., Бессонов А.С., Мошкин В.В., Папуловский В.Ф. *LabVIEW. Практикум по основам измерительных технологий*. Москва, ДМК-пресс, 2011, 301 с.
- [5] Трэвис Дж., Кринг Дж. *LabVIEW для всех*. Москва, ДМК-пресс, 2010, 123 с.
- [6] Федосов В.П., Нестеренко А.К. *Цифровая обработка сигналов в LabVIEW*. Москва, ДМК-пресс, 2011, 455 с.



- [7] Сергиенко А.М. VHDL для проектирования вычислительных устройств. Киев, ООО «ТИД «ДС», 203, 208 с.
- [8] Комолов Д., Золотуха Р. Использование микросхем специальной памяти для обеспечения защиты ПЛИС FPGA от копирования. URL: [http://www.kite.ru/articles/plis/2008\\_12\\_24.php](http://www.kite.ru/articles/plis/2008_12_24.php) (дата обращения 12.09.2018).
- [9] Бибило П.Н. *Основы языка VHDL*. Москва, СОЛОН-Р, 2002, 224 с.

**Черданова Екатерина Максимовна** — студентка кафедры «Защита информации», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

**Мамченко Елизавета Андреевна** — студентка кафедры «Защита информации», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

---

**ESTIMATION OF MEMORY USAGE OF THE FIELD PROGRAMMABLE GATE ARRAYS BY SIMPLE ALGORITHMS SUCH AS THE FAST FOURIER TRANSFORM USING THE LABVIEW FPGA MODULE**

**E.M. Cheredanova**

pankooova@mail.ru

SPIN-code: 1619-5499

**E.A. Mamchenko**

liza.98.98@mail.ru

SPIN- code: 2887-3715

**Bauman Moscow State Technical University, Moscow, Russian Federation**

---

**Abstract**

*The aim of the work is to study the workload of field programmable gate arrays (FPGAs) by standard algorithms for further system optimization by shifting some functions to the controller. For these purposes, the Fast Fourier Transform (FFT) algorithm on the FPGA is implemented in the LabVIEW application software environment using the LabVIEW FPGA software module. The created application consists of three modules: a separate virtual instrument made on the FPGA VI, a virtual instrument on the real-time controller (Real Time VI) and a virtual instrument on the personal computer with the Windows operating system (Host VI). As a result of the research, an assessment of the efficiency of memory usage on the FPGA using the analysis of the report on the compilation results of a binary file when replacing the FPGA embedded software using the LabVIEW FPGA module was made, which allows to estimate the amount of FPGA memory occupied by the simplest algorithms like FFT.*

**Keywords**

*The fast Fourier transform, field programmable gate arrays, FPGA, LabVIEW, virtual instrument, binary file*

Received 06.11.2018

© Bauman Moscow State Technical University, 2018

---

**References**

- [1] Turygin I.G. Metod vybora programmiruemykh logicheskikh integral'nykh skhem na osnove tselevogo funktsionala pri proektirovanii ustroystv tsifrovoy obrabotki informatsii. Diss. kand. tekhn. nauk [Method for choosing FPGA based on objective functional in engineering] digital information processing devices. Kand. tech. sci. diss.]. Penza, 2014, 134 p.
- [2] Galanina N.A. Metody i vychislitel'nye ustroystva tsifrovoy obrabotki signalov v sisteme ostatochnykh klassov. Avtoref. diss. dok. tekhn. nauk [Methods and computing unit of digital information processing in residue number system. Abs. kand. tech. sci. diss.]. Kazan', 2010, 32 p.
- [3] Avyashkieva N.S. Algoritm bystrogo preobrazovaniya Fur'ye v sisteme ostatochnykh klassov [Fast Fourier transform algorithm in residue number system]. *Tekhnicheskie nauki — ot teorii k praktike. Sb. st. po mater. V mezhd. nauch.-prakt. konf.* [Technical sciences — from theory to practice. Proc. V int. sci.-pract. conf.]. Novosibirsk, SibAK publ., 2011, pp. 79–83.
- [4] Batovrin V.K., Bessonov A.S., Moshkin V.V., Papulovskiy V.F. LabVIEW. Praktikum po osnovam izmeritel'nykh tekhnologiy [LabVIEW. Practicum based on measuring technologies]. Moscow, DMK-press publ., 2011, 301 p.

- [5] Travis J., Kring J. LabVIEW for everyone: graphical programming made easy and fun. Prentice Hall, 1040 p. (Russ. ed.: LabVIEW dlya vsekh. Moscow, DMK-press publ., 2010, 123 p.)
- [6] Fedosov V.P., Nesterenko A.K. Tsifrovaya obrabotka signalov v LabVIEW [Digital signal processing in LabVIEW]. Moscow, DMK-press publ., 2011, 455 p.
- [7] Sergienko A.M. VHDL dlya proektirovaniya vychislitel'nykh ustroystv [VHDL for designing of computing devices]. Kiev, OOO "TID "DS»" publ., 203, 208 p.
- [8] Komolov D., Zolotukho R. Ispol'zovanie mikroskhem spetsial'noy pamyati dlya obespecheniya zashchity PLIS FPGA ot kopirovaniya [Using special memory microcircuits for providing FPGA security from copying]. Available at: [http://www.kite.ru/articles/plis/2008\\_12\\_24.php](http://www.kite.ru/articles/plis/2008_12_24.php) (accessed 12 September 2018).
- [9] Bibilo P.N. Osnovy yazyka VHDL [Fundamentals of VHDL language]. Moscow, SOLON-R publ., 2002, 224 p.

**Cheredanova E.M.** — student, Department of Information Security, Bauman Moscow State Technical University, Moscow, Russian Federation.

**Mamchenko E.A.** — student, Department of Information Security, Bauman Moscow State Technical University, Moscow, Russian Federation.