

ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ

Научная статья

УДК 004.31

DOI: [10.34759/trd-2021-120-13](https://doi.org/10.34759/trd-2021-120-13)

Метод и устройство расположения задач в реконфигурируемых вычислительных системах

Илья Игоревич Масюков

Юго-Западный государственный университет, ЮЗГУ, Курск, Россия

ilmas46ru@gmail.com

Аннотация. В статье рассматривается разработанная модель, алгоритм и устройство расположения задач в реконфигурируемых вычислительных системах позволяющие снизить время поиска конфигурации. Модель описывает построение устройства и четыре условия расположения вершин между ПЛИС, позволяющих уменьшить вычислительную сложность алгоритма. Разработанный алгоритм в совокупности процедур выбора, хранения и присвоения допускает реализацию на текущих аппаратных средствах. В ходе экспериментальных исследований разработанного устройства проводился анализ быстродействия и полученной конечной конфигурации, показавший, что время работы алгоритма зависит от количества вершин. Результаты исследования позволяют сделать вывод, что разработанное устройство быстрее аналогичных программных средств в 5,17 раза. Таким образом, о преимуществе аппаратной реализации представленного метода и алгоритма.

Ключевые слова: реконфигурируемая вычислительная система, ПЛИС, конфигурация, алгоритм, граф

Для цитирования: Масюков И.И. Метод и устройство расположения задач в реконфигурируемых вычислительных системах // Труды МАИ. 2021. № 120. DOI: [10.34759/trd-2021-120-13](https://doi.org/10.34759/trd-2021-120-13)

INFORMATICS, COMPUTATION ENGINEERING AND MANAGEMENT

Original article

Method and device for arranging tasks in reconfigurable computing systems

Иля I. Masyukov

South-Western State University, Kursk, Russia

ilmas46ru@gmail.com

Abstract. In view of the growing complexity of the problems being solved, reconfigurable computing systems (RCS) have become widespread in various branches such as industry, aviation, instrument making, etc. The RCS distinctive aspect consists in the ability of the internal architecture changing for the task being solved, allowing productivity and energy efficiency increasing. The high degree of integration and hardware configuration allowed FPGAs application as the main computational element. When solving the information graph on the RCS, in view of the employed and available resources of the reconfigurable field, it is being divided into subgraphs, for which a configuration, describing the tasks locating among the FPGAs is being composed. Due to the complexity of the

location search problem, which is NP-complex, its execution on a host computer operating in real time seems impossible. The existing approaches are either inapplicable for the DCS configuration search, or being solved programmatically. Thus, the development of the tools reducing the search time for the RCS configuration is up-to-date. An algorithm and a model, including the problem of the set covering, were created when developing a unit for the tasks location in the RCS. The model describes the unit construction, and four criteria. These criteria are being based on the ideas of tasks selecting from an adjacency matrix based on the number of links, the intensity of exchange between them and the occupied and available resources for location, allowing reduce the number of enumeration options, which made it possible to reduce the computational complexity of the algorithm. The algorithm developed on the basis of the model, in the set of selection, storage and assignment procedures, can be implemented on the current hardware. In the course of experimental studies of the tasks arrangement in the RCS, an analysis of the performance and the resulting final configuration was performed. It is shown that the developed unit is 5.17 times faster than the similar software. Thus, the inference can be drawn on the hardware implementation of the presented method and algorithm advantage. The developed unit is applicable in the RCS operating in real time, due to its productivity increase.

Keywords: reconfigurable computing system, FPGA, configuration, algorithm, graph

For citation: Masyukov I.I. Method and device for arranging tasks in reconfigurable computing systems. *Trudy MAI*, 2021, no. 120. DOI: [10.34759/trd-2021-120-13](https://doi.org/10.34759/trd-2021-120-13)

Введение

Суперкомпьютеры применяются для решения вычислительно сложных задач криптографии, цифровой обработки сигналов, радиолокации и т.п. Но из-за кластерной организации большинства суперЭВМ их максимальная эффективность достигается при решении задач определенных классов [1]. Для устранения особенностей вычислителей с кластерной организацией был разработан подход с реконфигурируемой архитектурой, позволяющий изменять внутреннюю организацию устройства под решаемую задачу [2]. Развитие технологий программируемых логических интегральных схем (ПЛИС) позволило создать реконфигурируемые вычислительные системы (РВС) высокой степени интеграции, объединяемые в единое вычислительное поле [3].

Вычислительная задача представляется в виде информационного графа, в котором вершины обозначают задачи, их вес определяет занимаемые ресурсы, взвешенные ребра – интенсивность обмена. Составление плана конфигурации – это поиск отношения вершин подграфа задачи к микросхемам ПЛИС. Однако, из-за технических ограничений существующих РВС и постоянного роста требований решаемых задач к вычислительным ресурсам, реализовать информационный граф в полном объеме не представляется возможным [4]. Следовательно, для решения задачи происходит разделение графа на подграфы, последовательно реализуемые на реконфигурируемом поле, из-за чего возникает пауза между окончанием работы

системы для одного подграфа задачи и составлением нового расположения следующего подграфа [5].

Таким образом, целью решения задачи поиска конфигурации для подграфа информационного графа в РВС заключается в минимизации времени составления плана расположения.

Точные алгоритмы решения данной задачи, основанные на идеях ветвей и границ [6], отсечения [7], перебора L-классов [8] дают оптимальное расположение задач на ПЛИС, но их решение требует полиномиального времени, что неприменимо для систем высокой готовности.

Эвристические алгоритмы, например, генетический [6], муравьиной колонии [9] и т.п. реализуются программным способом, что не позволяет достичь максимального снижения времени составления плана расположения [10-13].

Таким образом, существует необходимость разработки новых подходов расположения задач в РВС, позволяющих добиться снижения времени поиска конфигурации и тем самым повысить вычислительную мощность [14-17].

Модель, метод и устройство расположения задач в РВС

Взаимодействие задач представим как взвешенный граф $G = \langle X, E \rangle$, где $x_q \in X$ – номер задачи, $e_{ij} \in E$ – обмен между ними. Ребра сведены в матрицу смежности $M = |e_{ij}|_{N \times N}$, где $N = |X|$, $i, j = \overline{1, n}$ [10].

Поле для расположения задач является множество $A = \{S_1, S_2, \dots, S_i\}$ с подмножествами S , описывающими ПЛИС. Размер ПЛИС задается C_i , где $i \in A$.

S принадлежат вершины графа G :

$$S_i = \{x_n, \dots, x_m\},$$

где $x \in G$, а $S_i \in A$.

Занимаемое пространство программой на ПЛИС опишем множеством $W = \{w_1, w_2, \dots, w_n\}$, где n равно количеству вершин.

Для достижения минимальной коммутационной задержки в системе были предложены условия размещения задач между ПЛИС:

1. Задачи, имеющие максимальное число связей с другими задачами, должны совместно располагаться на одной ПЛИС:

$$x_n \in S, \text{ если } d(x_n) \rightarrow \max. (1)$$

2. Задачи, максимально связанные между собой, должны располагаться на одной ПЛИС:

$$x_n \in S \text{ и } x_m \in S, \text{ если } |e_{ij}| = x_n, x_m \rightarrow \max. (2)$$

3. Задачи, имеющие связь только с заданной, должны располагаться на одной ПЛИС:

$$x_n \in S \text{ и } x_m \in S, \text{ если } e_{ij} = x_n, x_m \text{ и } d(x_m) = 1, (3)$$

где $x_n, x_m \in X, e_{ij} \in E$.

4. Сумма весов располагаемых вершин, не должна превышать максимально возможного размера ПЛИС:

$$\sum_{W \in S} f(S) \leq C_i. (4)$$

Тогда задача расположения подграфов информационного графа в РВС сводится к нахождению множества A , в котором $X \in A$ и выполняются условия (1)-(4).

Отличительной особенностью разработанной модели от задачи о покрытии множества является введение дополнительных условий расположения, описывающих формирование отношения задач к ПЛИС и позволяющих уменьшить вычислительную сложность алгоритма.

На основании предложенной выше модели был разработан алгоритм расположения задач в РВС:

1. Выбираем вершину с максимальным числом соединений. Далее по тексту обозначена как «опорная» (формула 1).

2. Выбираем смежную «опорной» вершину с максимальным весом ребра (формула 2).

3. Находим для выбранных на 1 и 2 шаге вершин смежные висячие вершины (формула 3).

4. Сумма весов вершин не превышает размера ПЛИС (формула 4)? Да – п. 5, иначе п. 7.

5. Присваиваем номера выбранных вершин на 1-3 шаге текущей ПЛИС.

6. Удаляем из матрицы смежности все связи с данными вершинами. Переходим к п. 8.

7. Первая итерация? Да – добавляем вершину с максимальным весом ребра в черный список и переходим к п.4. Иначе берём новую ПЛИС и переходим к п.4.

8. Если остались смежные с «опорной» вершины – переходим к п. 2. Иначе удаляем все связи с опорной вершиной из матрицы смежности, очищаем черный список вершин и переходим к п. 9.

9. Если остались вершины со связями – п. 1, иначе – п. 10.

10. Все вершины расположены? Да – п.11, иначе располагаем изолированные вершины в порядке возрастания номера, проверяя суммарный вес, который не должен превышать максимального размера ПЛИС.

11. Конец алгоритма.

Особенность разработанного алгоритма заключается в выборе вершин по количеству связей и интенсивности обмена между ними, с последующим расположением на ПЛИС, позволившая снизить вычислительную сложность до $O(f(N^2))$ для матрицы смежности, размером N .

Основываясь на представленный алгоритм, была разработана структурно-функциональная схема устройства расположения задач в РВС, представленная на рис. 1.

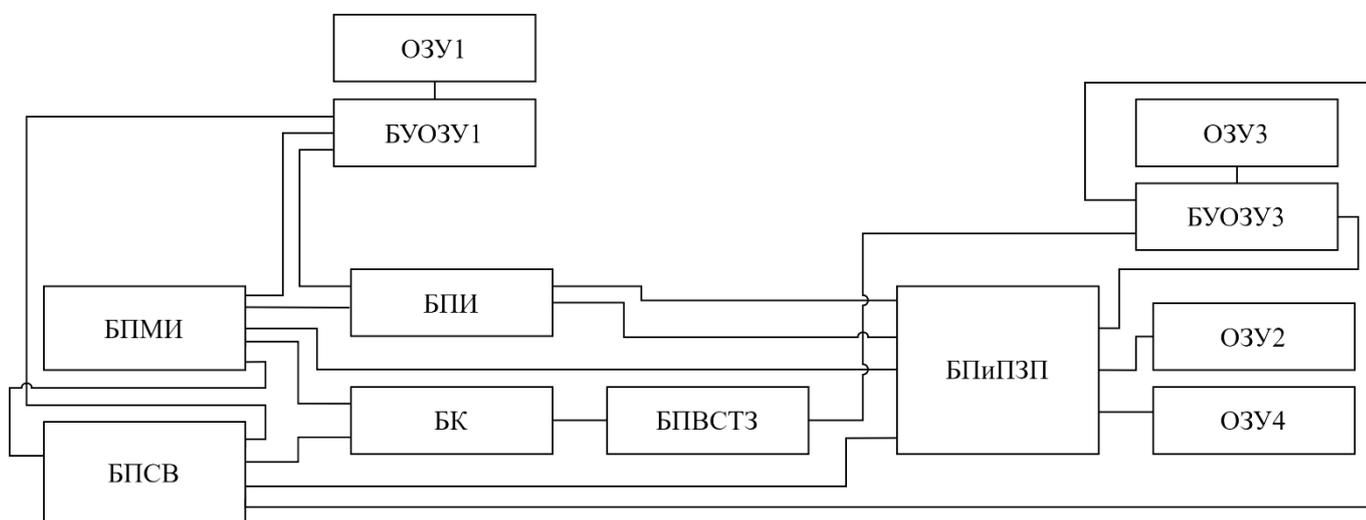


Рис. 1. Структурно-функциональная схема устройства расположения задач в РВС

Для работы устройства используются четыре оперативно запоминающих устройства (ОЗУ). В ОЗУ1 хранится матрица смежности графа задач, в ОЗУ2 – веса

вершин, ОЗУ3 служит местом промежуточного хранения данных, в ОЗУ4 хранятся вершины с присвоенными им номерами ПЛИС.

Также функционирование устройства обеспечивается следующими блоками:

- поиска максимальной инцидентности (БПМИ) – находит в матрице смежности графа задачи вершину с максимальным количеством смежностей ("опорную");

- поиска смежной вершины (БПСВ) – определяет для "опорной" смежную вершину с максимальным весом ребра;

- подсчета инцидентностей (БПИ) – рассчитывает значение инцидентности "опорной" вершины в ходе работы алгоритма;

- поиска смежной вершины только заданной (БПВСТЗ);

- контроля (БК), управляющий БПВСТЗ;

- управления ОЗУ1 (БОЗУ1) – организывает доступ блоков к ОЗУ1, хранит массив «черного списка» вершин, удаляет сведения о вершинах в ОЗУ1;

- управления ОЗУ3 (БУОЗУ3) – позволяет работать с ОЗУ3 как с FIFO с сохранением данных, позволяя упростить структуру устройства. Также осуществляет функции очистки и добавления вершин;

- поиска и присвоения задач ПЛИС (БПиПЗП) – организывает работу устройства и присваивает задачи ПЛИС, основываясь на условиях суммы весов, считываемых из ОЗУ2.

На основании предложенной структурно-функциональной схемы (рис. 1) был разработан код на языке Verilog [11-13] с использованием ПО Quartus. В качестве

ПЛИС выбрана 10M16SAE1448G семейства MAX10. Данная ПЛИС состоит из 15840 логических элементов, включающих четырех-входные LUT, 562176 бит встроенной памяти, 90 встроенных умножителей 9-ти битовых слов. [14-17].

При компиляции разработанного кода ПО Quartus выделило следующие ресурсы ПЛИС:

- общее количество логических элементов – 1437;
- общее количество комбинационных функций – 1125;
- использование выделенных логических регистров – 750;
- общее количество битов памяти – 10240.

Результаты и их обсуждение

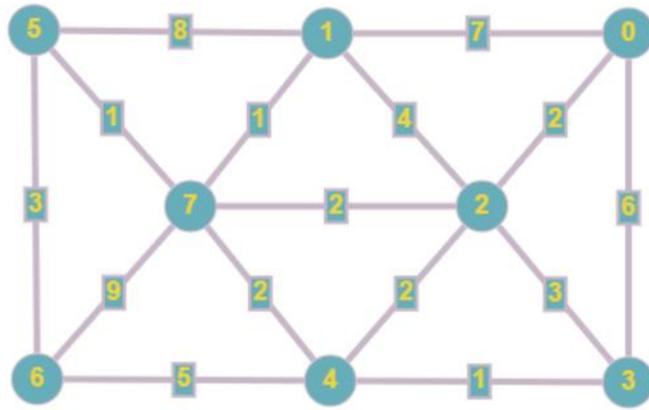
Время поиска конфигурации проводилось для следующих графов: планарный, полный, звезда с соединенными кольцом внешними вершинами, двудольный [18-19].

Примеры графов представлены на рис. 2. Размер ПЛИС во всех случаях принят за 20.

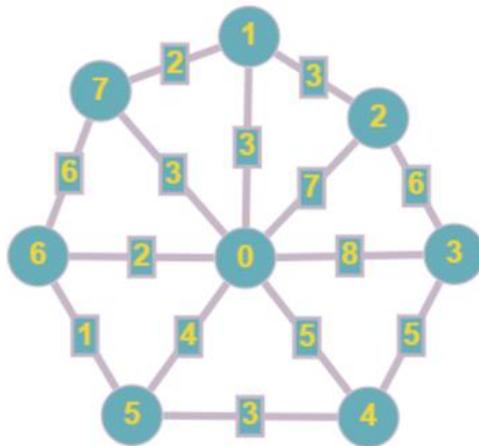
Значение весов каждой вершины для всех типов графов указаны в табл. 1.

Таблица 1. Значение весов вершин в графе

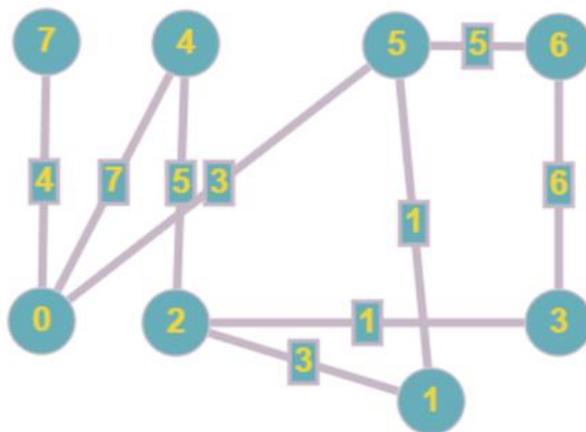
Номер вершины	1	2	3	4	5	6	7	8
Вес вершины	8	5	10	10	4	2	7	7



а)



б)



в)

Рис. 2. Виды графов, для которых проводились исследования: **а** - планарный граф; **б** - граф типа звезда с соединенными кольцом внешними вершинами; **в** - двудольный граф

Моделирование работы разработанного устройства проводились в ПО ModelSim, пример работы которого представлен на рис. 3-5 [20-21]. Сигналом окончания работы устройства является выставление логической единицы на сигнале `algorithm_done`. Тактовая частота работы устройства составила – 100 МГц.

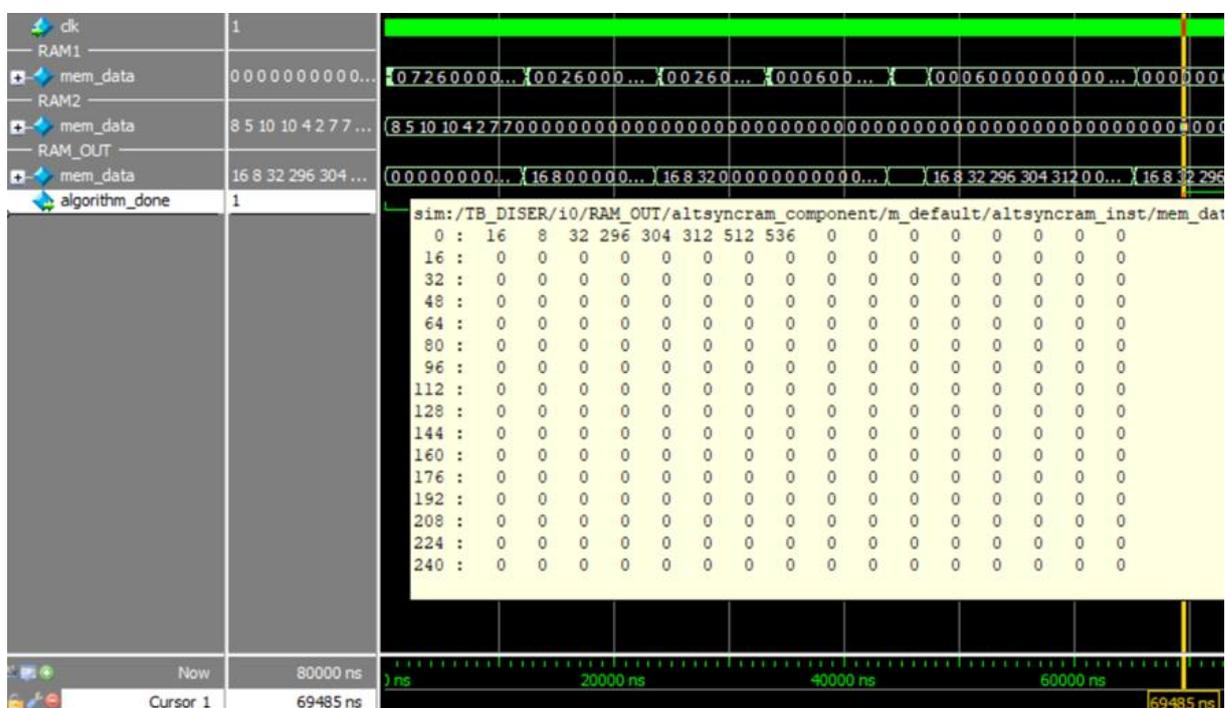


Рис. 3. Результат работы устройства поиска конфигурации ПЛИС системы для планарного графа

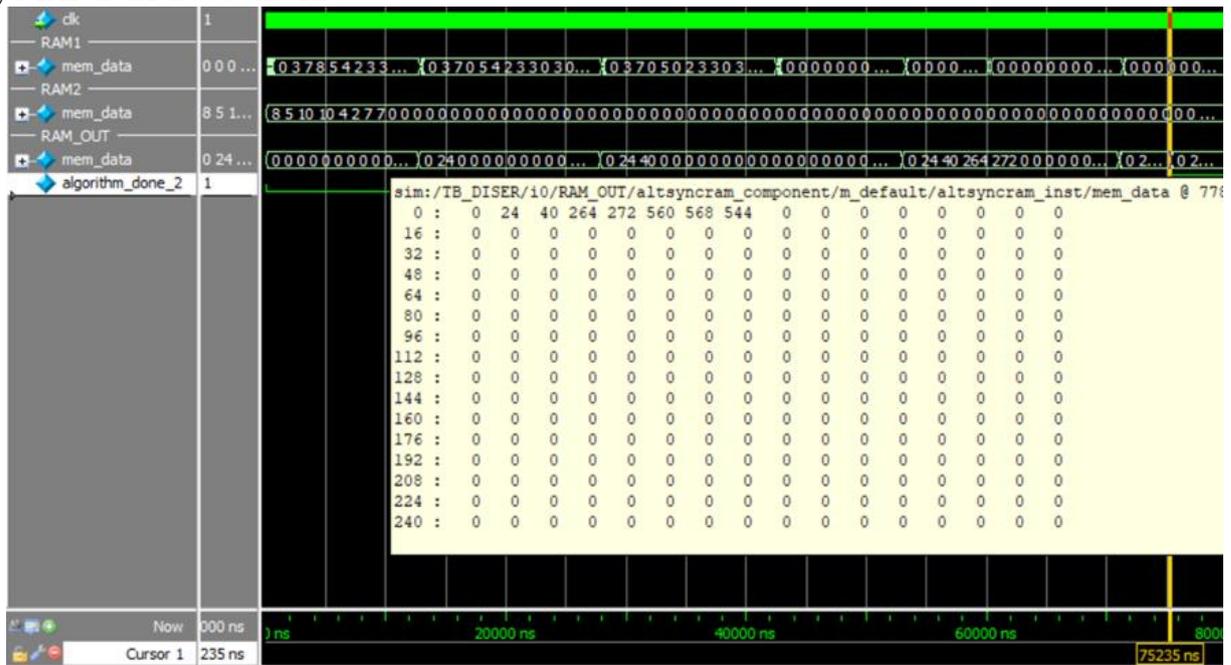


Рис. 4. Результат работы устройства поиска конфигурации ПЛИС системы для графа звезда, с соединенными кольцом внешними вершинами

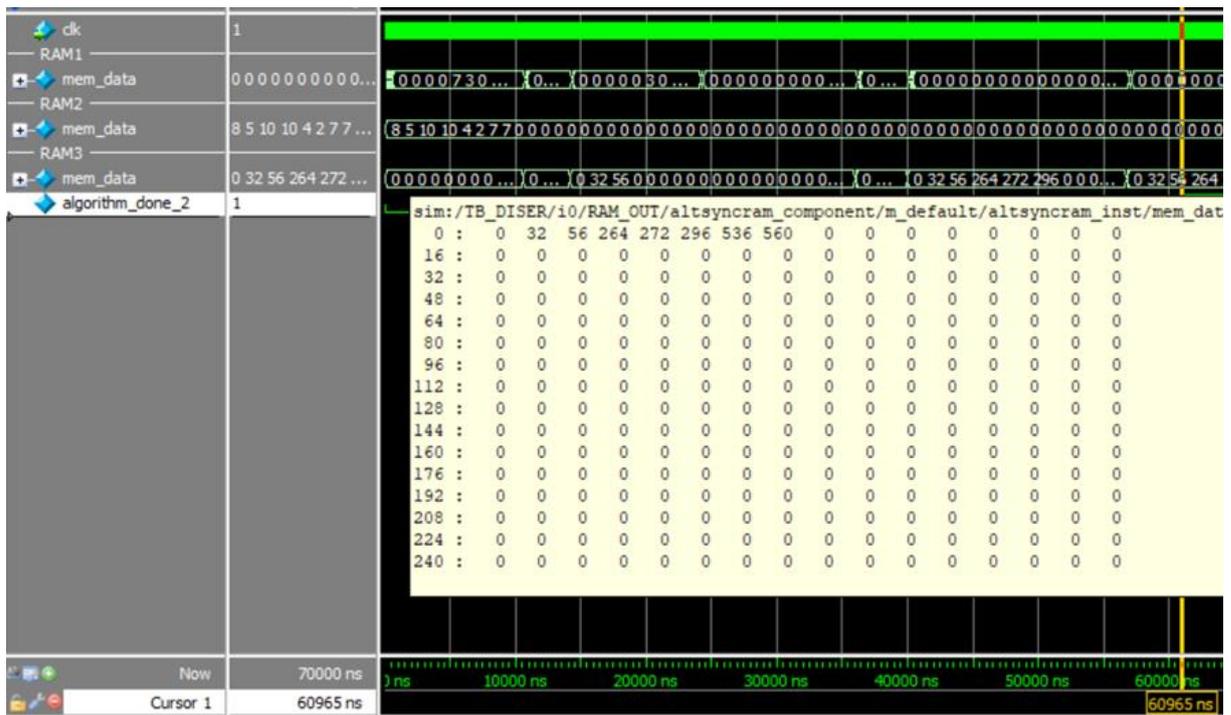


Рис. 5. Результат работы устройства поиска конфигурации ПЛИС системы для двудольного графа

На рисунках 3-5 слева обозначены названия цепей, соответствующие сигналу тактирования, память хранения матрицы смежности, значения весов вершин и конечной конфигурации, а также сигнала окончания работы. Справа – изменение во времени соответствующих цепей. Также выделены значения памяти, в которую записывается конечная конфигурация.

Результаты расположения вершин между ПЛИС показаны в табл. 2.

Таблица 2. Расположение задач между ПЛИС

Тип графа	Распределение задач на ПЛИС		
	ПЛИС №1	ПЛИС №2	ПЛИС №3
Планарный граф	2, 1, 4	5, 6, 7	0, 3
Звезда, с соединенными кольцом внешними вершинами	0, 3, 5	1, 2	6, 7, 4
Двудольный граф	0, 4, 7	1, 2, 5	3, 6

Результаты исследования устройства расположения задач в ПЛИС системах приведены в таблице 3.

Таблица 3. Результаты работы устройства расположения задач в РВС

Тип графа	Планарный	Звезда, с соединенными кольцом внешними вершинами	Двудольный
Время расчета конфигурации, мкс	69,5	75,2	61
Общее кол-во связей в графе	56	58	35
Кол-во связей, расположенных на модуле ПЛИС	25	21	21
Кол-во связей, расположенных на внешнем коммутаторе	31	37	14
Отношение связей на ПЛИС к связям на коммутаторе	0,81	0,57	1,5

Сравнение результатов работы разработанного устройства с программными аналогами жадного и генетического алгоритмов, представлен в табл. 4.

Таблица 4. Сравнение разработанного устройства с программными аналогами

	Время, мкс
Устройство составления плана конфигурации	68,6
Генетический алгоритм [22]	254,2
Жадный алгоритм [22]	101,141

Из анализа полученных результатов можно сделать вывод, что разработанное устройство расположения задач в РВС эффективнее аналогов в 5,17 раза.

Выводы

В работе проведено исследование устройства расположения задач в реконфигурируемых вычислительных системах для трех видов графов. Из анализа полученных экспериментальным путем данных, сделан вывод, что устройство уменьшает время расположения задач относительно существующих аналогов в 5,17 раза.

Список источников

1. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. - Санкт-Петербург: БХВ-Петербург, 2015. – 608 с.
2. Гузик В.Ф., Каляев И.А., Левин И.И. Реконфигурируемые вычислительные системы. - Таганрог: Южный федеральный университет, 2016. - 472 с.
3. Каляев И.А., Левин И.И., Семерников Е.А., Шмойлов В.И. Реконфигурируемые мультиконвейерные вычислительные структуры. - Ростов на Дону: ЮНЦ РАН, 2008. - 320 с.
4. Масюков И.И., Борзов Д.Б., Титов Д.В., Соколова Ю.В. Математическая модель и аппаратно-ориентированный алгоритм планирования размещения программ в

<http://trudymai.ru/published.php?ID=159791>. DOI: [10.34759/trd-2021-119-13](https://doi.org/10.34759/trd-2021-119-13)

5. Борзов Д.Б., Кошелев М.А., Соколова Ю.В. Моделирование реконфигурируемой системы реального времени с учетом уменьшения временных затрат обработки информации // Труды МАИ. 2021. № 117. URL:

<http://trudymai.ru/published.php?ID=156284>. DOI: [10.34759/trd-2021-117-13](https://doi.org/10.34759/trd-2021-117-13)

6. Коновалов И.С., Остапенко С.С., Кобак В.Г. Сравнение эффективности работы точных и приближенных алгоритмов для решения задачи о покрытии множества // Вестник Донского государственного технического университета. 2017. № 3(90). С. 137–144.

7. Добряков В.А., Енгальчев А.Н., Назаров А.В. Начальное размещение базовых элементов комплементарных металл-окисел-полупроводниковых больших интегральных схем методом случайных назначений // Труды МАИ. 2014. № 72. URL:

<http://trudymai.ru/published.php?ID=47562>

8. Заозерская Л.А., Колоколов А.А. Исследование и решение двухкритериальной задачи о покрытии множества // Проблемы информатики. 2009. № 1(2). С. 14–23.

9. Лебедев Б.К., Лебедев О.Б. Покрытие на основе методов роевого интеллекта // Проблемы разработки перспективных микро- и наноэлектронных систем. 2016. № 1. С. 187-194.

10. Панкратов А.В., Якимов В.Л., Маковский В.Н. Анализ избыточности битовой последовательности для проектов программируемых логических интегральных схем // Труды МАИ. 2015. № 82. URL: <http://trudymai.ru/published.php?ID=58828>

Труды МАИ. 2021. Выпуск № 120
Trudy MAI. 2021. Issues no. 120

11. Борзов Д.Б., Басов Р.Г., Титов В.С., Соколова Ю.В. Устройство планирования загрузки процессоров в мультипроцессорных системах критического назначения //

Труды МАИ. 2020. № 115. URL: <http://trudymai.ru/published.php?ID=119942>. DOI: [10.34759/trd-2020-115-14](https://doi.org/10.34759/trd-2020-115-14)

12. Соловьев В.В. Основы языка проектирования цифровой аппаратуры Verilog. - М.: Горячая Линия-Телеком, 2021. - 364 с.

13. Моисеев Д.В., Чинь В.М., Мозолев Л.А., Моисеева С.Г., Фам С.К. Маршрутизация полета легкого беспилотного летательного аппарата в поле постоянного ветра на основе решения разновидностей задачи коммивояжера // Труды МАИ. 2015. № 79. URL: <http://trudymai.ru/published.php?ID=55782>

14. Зотов В.Ю. Особенности архитектуры нового поколения ПЛИС с архитектурой FPGA фирмы Xilinx // Компоненты и технологии. 2010. № 12. С. 17-24.

15. Pedroni V.A. Digital Electronics and Design with VHDL, 1st edition, Burlington, Morgan Kaufmann, 2008, 717 p.

16. Ledin J. Architecting High-Performance Embedded Systems: Design and build high-performance real-time digital systems based on FPGAs and custom circuits, Birmingham, Packt Publishing, 2021, 376 p.

17. Deschamps J., Sutter D. G., Cantó E. Guide to FPGA Implementation of Arithmetic Functions, Berlin, Springer, 2012, 472 p.

18. Оре О. Теория графов. – М.: Либроком, 2009. – 354 с.

19. Уилсон Р. Введение в теорию графов. – М.: Вильямс, 2020. - 240 с.

20. Матафонов Д.Е. Создание и обработка маршрутизатора в стандарте SpaceWire на отечественной программируемой логической интегральной схеме // Труды МАИ.

2018. № 103. URL: <http://trudymai.ru/published.php?ID=100780>

21. Кордовер К.А., Жданов А.А., Данилов А.М. Универсальный блок управления массивом запоминающих устройств наземного отладочного комплекса // Труды

МАИ. 2013. № 65. URL: <http://trudymai.ru/published.php?ID=35985>

22. Коновалов И.С., Фатхи В.А., Кобак В.Г. Применение генетического алгоритма для решения задачи покрытия множеств // Вестник Донского государственного

технического университета. 2016. № 3(86). С. 125-132.

References

1. Voevodin V.V., Voevodin V.I. *Parallel'nye vychisleniya* (Parallel computing), Saint Peterburg, BKhV-Peterburg, 2015, 608 p.

2. Guzik V.F., Kalyaev I.A., Levin I.I. *Rekonfiguriruemye vychislitel'nye sistemy* (Reconfigurable Computing Systems), Taganrog, Yuzhnyi federal'nyi universitet, 2016, 472 p.

3. Kalyaev I.A., Levin I.I., Semernikov E.A., Shmoilov V.I. *Rekonfiguriruemye mul'tikonveiernye vychislitel'nye struktury* (Reconfigurable multicore computational structures), Rostov na Donu, YuNTs RAN, 2008, 320 p.

4. Masyukov I.I., Borzov D.B., Titov D.V., Sokolova Yu.V. *Trudy MAI*, 2021, no. 119. URL <http://trudymai.ru/eng/published.php?ID=159791>. DOI: [10.34759/trd-2021-119-13](https://doi.org/10.34759/trd-2021-119-13)

5. Borzov D.B., Koshelev M.A., Sokolova Yu.V. *Trudy MAI*, 2021, no. 117. URL: <http://trudymai.ru/eng/published.php?ID=156284>. DOI: [10.34759/trd-2021-117-13](https://doi.org/10.34759/trd-2021-117-13)
6. Konovalov I.S., Ostapenko S.S., Kobak V.G. *Vestnik Donskogo gosudarstvennogo tekhnicheskogo universiteta*. 2017, no. 3(90), pp. 137–144.
7. Dobryakov V.A., Engalychev A.N., Nazarov A.V. *Trudy MAI*, 2014, no. 72. URL: <http://trudymai.ru/eng/published.php?ID=47562>
8. Zaozerskaya L.A., Kolokolov A.A. *Problemy informatiki*, 2009, no. 1(2), pp. 14–23.
9. Lebedev B.K., Lebedev O.B. *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh system*, 2016, no. 1, pp. 187-194.
10. Pankratov A.V., Yakimov V.L., Makovskii V.N. *Trudy MAI*, 2015, no. 82. URL: <http://trudymai.ru/eng/published.php?ID=58828>
11. Borzov D.B., Basov R.G., Titov V.S., Sokolova Yu.V. *Trudy MAI*, 2020, no. 115. URL: <http://trudymai.ru/eng/published.php?ID=119942>. DOI: [10.34759/trd-2020-115-14](https://doi.org/10.34759/trd-2020-115-14)
12. Solov'ev V.V. *Osnovy yazyka proektirovaniya tsifrovoi apparatury Verilog* (Fundamentals of the Verilog digital hardware design language), Moscow, Goryachaya Liniya-Telekom, 2021, 364 p.
13. Moiseev D.V., Chin' V.M., Mozolev L.A., Moiseeva S.G., Fam S.K. *Trudy MAI*, 2015, no. 79. URL <http://trudymai.ru/eng/published.php?ID=55782>
14. Zotov V.Yu. *Komponenty i tekhnologii*, 2010, no. 12, pp. 17-24.
15. Pedroni V.A. *Digital Electronics and Design with VHDL*, 1st edition, Burlington, Morgan Kaufmann, 2008, 717 p.

16. Ledin J. *Architecting High-Performance Embedded Systems: Design and build high-performance real-time digital systems based on FPGAs and custom circuits*, Birmingham, Packt Publishing, 2021, 376 p.
17. Deschamps J., Sutter D. G., Cantó E. *Guide to FPGA Implementation of Arithmetic Functions*, Berlin, Springer, 2012, 472 p.
18. Ore O. *Teoriya grafov* (Graph theory), Moscow, Librokom, 2009, 354 p.
19. Uilson R. *Vvedenie v teoriyu grafov* (Introduction to graph theory), Moscow, Vil'yams, 2020, 240 p.
20. Matafonov D.E. *Trudy MAI*, 2018, no. 103. URL: <http://trudymai.ru/eng/published.php?ID=100780>
21. Kordover K.A., Zhdanov A.A., Danilov A.M. *Trudy MAI*, 2013, no. 65. URL: <http://trudymai.ru/eng/published.php?ID=35985>
22. Konovalov I.S., Fatkhi V.A., Kobak V.G. *Vestnik Donskogo gosudarstvennogo tekhnicheskogo universiteta*, 2016, no. 3(86), pp. 125-132.

Статья поступила в редакцию 15.09.2021; одобрена после рецензирования 24.09.2021; принята к публикации 22.10.2021.

The article was submitted 15.09.2021; approved after review 24.09.2021; accepted for publication 22.10.2021.