УДК 621.396.06 Научная статья

https://doi.org/10.32603/2071-8985-2026-19-9-54-63

Алгоритм трассировки ячеек БИС, учитывающий особенности топологической реализации

П. М. Ширяев

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина), Санкт-Петербург, Россия

pshir2000z@gmail.com

Аннотация. Предложен алгоритм трассировки для инвариантного проектирования ячеек БИС, ключевая особенность которого – оперирование группами точек подключения вместо отдельных координат. Алгоритм поддерживает гибкую настройку через систему весовых коэффициентов для каждого направления в каждом технологическом слое, позволяя адаптировать метод под различные проектные сценарии. Экспериментальная проверка показала, что введение мягких ограничений на направления проводников улучшает итоговые метрики. Полученные результаты подтверждают применимость предложенного подхода для автоматизированной генерации ячеек в условиях инвариантных технологических требований.

Ключевые слова: алгоритм трассировки, размещение и трассировка, плотноупакованное проектирование топологии интегральных схем; технологическая инвариантность

Для цитирования: Ширяев П. М. Алгоритм трассировки ячеек БИС, учитывающий особенности топологической реализации // Изв. СПбГЭТУ «ЛЭТИ». 2026. Т. 19, № 9. С. 54–63. doi: 10.32603/2071-8985-2026-19-9-54-63.

Original article

Routing Algorithm for VLSI Cells Accounting for Topological Implementation Specifics

P. M. Shiryaev

Saint Petersburg Electrotechnical University, Saint Petersburg, Russia

pshir2000z@gmail.com

Abstract. This paper presents a routing algorithm for invariant VLSI cell design, characterized by the use of connection point groups instead of individual coordinates. The algorithm allows flexible tuning via a system of directional weight coefficients for each technological layer, enabling adaptation to various design scenarios. Experimental results demonstrate that applying soft constraints on wire directions yields better performance metrics. These findings confirm the suitability of the proposed method for automated cell generation under invariant technological constraints.

Keywords: routing algorithm, placement and routing, compact VLSI layout design, technological invariance

For citation: Shiryaev P. M. Routing Algorithm for VLSI Cells Accounting for Topological Implementation Specifics // LETI Transactions on Electrical Engineering & Computer Science. 2026. Vol. 19, no. 9. P. 54–63. doi: 10.32603/2071-8985-2026-19-9-54-63.

Введение. Проектирование современных вычислительных систем и электронных модулей сегодня характеризуется высокой сложностью. Увеличение плотности компоновки, ужесточение технологических норм и рост требований к производительности делают ручные методы разработки непрактичными. В таких условиях автоматизация проектирования становится необходимым этапом, без которого невозможно обеспечить ни сроки, ни качество конечного устройства.

Один из ключевых процессов в автоматизированном проектировании – задача трассировки. На

 © Ширяев П. М., 2026

этом этапе формируются физические соединения между компонентами, от которых напрямую зависят задержки сигналов и как следствие быстродействие устройства.

В зависимости от требований на данном этапе приоритет может отдаваться тем или иным характеристикам из-за чего требуются алгоритмы, которые могут адаптироваться к изменяющимся условиям без полной перестройки логики работы.

В данной статье предложен алгоритм трассировки, предназначенный для автоматизации этапа генерации топологии ячейки в процессе инвариантного проектирования. В статье описана структура алгоритма, изложены основные принципы его работы и приведены результаты проверки на типовых тестовых примерах.

Постановка задачи. Задача трассировки заключается в поиске допустимых геометрических путей для электрических соединений между компонентами с учетом технологических ограничений и проектных правил. На уровне ячейки пути строятся между контактами транзисторов, шинами и внешними контактами ячейки.

Для данного этапа существует ряд обязательных требований [1]. Обязательна функциональная эквивалентность топологии исходной электрической схеме, что подразумевает полное завершение трассировки. Еще одно требование – это соответствие технологическим нормам, без соблюдения которых произвести устройство будет невозможно или экономически невыгодно.

Специфика трассировки на уровне ячеек, обусловленная технологическими процессами производства, накладывает ряд дополнительных ограничений на то, как должны быть организованы проводники в топологии [2]. Переходов между слоями следует избегать, поскольку каждый такой переход увеличивает сопротивление цепи, усложняет топологию и требует дополнительной площади. Не допускается выполнять подключение к затвору транзистора в его срединной части, что обусловлено технологическими нормами изготовления и требованиями к равномерности электрических характеристик. Слой поликремния может использоваться для формирования проводников, однако его применение ограничено из-за повышенного по сравнению с металлическими слоями удельного сопротивления.

Наконец, подключение к затвору транзистора может быть организовано с разных сторон, но для корректной работы транзистора достаточно обеспечить соединение только с одной стороны. Это позволит избежать лишних переходов со слоя металла на слой поликремния, что упрощает этап производства и процесс сжатия топологии.

Критерии качества трассировки. Качество полученного решения оценивается по набору количественных показателей.

Общая длина проводников выступает базовым критерием оптимизации, поскольку непосредственно определяет паразитную емкость и сопротивление межсоединений. Увеличение суммарной длины приводит к росту задержек распространения сигнала и повышению динамического энергопотребления, что особенно критично для высокочастотных и энергоэффективных схем. Кроме того, более длинные проводники занимают больше места в трассировочном пространстве, снижая плотность компоновки и усложняя прокладку последующих соединений.

Максимальная длина единичного проводника контролируется обособленно, так как именно наиболее протяженные соединения формируют критические пути, определяющие быстродействие всей ячейки. Превышение допустимой длины может привести к нарушению временных ограничений, сбоям в синхронных цепях или необходимости введения дополнительных буферов, что увеличивает площадь и энергозатраты. Поэтому алгоритм должен не только минимизировать средние значения, но и гарантировать, что ни один проводник не выйдет за установленные пределы [3].

Немаловажным критерием служит количество переходов между слоями (via). Количество переходов напрямую влияет на надежность и себестоимость изготовления. Каждый межслойный переход представляет собой потенциальное место отказа из-за технологических погрешностей металлизации, а также вносит дополнительное сопротивление и паразитную индуктивность в цепь. В плане трассировки, избыточное количество переходов усложняет топологию, требует соблюдения дополнительных зазоров и может создавать локальные зоны перегрузки, блокирующие прокладку соседних соединений.

Критерием может служить занимаемая ячейкой площадь. Именно площадь ячейки определяет итоговую плотность размещения на кристалле и, как следствие, экономическую эффективность производства. Чем компактнее ячейка, тем больше функциональных блоков можно разместить на одной пластине, снижая удельную стоимость каждого чипа. При этом уменьшение площади не должно достигаться за счет нарушения проектных

норм или ухудшения электрических характеристик, поэтому алгоритм ищет баланс между миниатюризацией и соблюдением всех ограничений.

Другой широко применяемый для больших схем критерий – это количественная оценка скоплений проводников [4], которая вводится как дополнительный метрический инструмент для анализа локальной перегруженности трассировочного пространства. Высокая плотность проводников в ограниченной области повышает риск нарушения минимальных зазоров, создает условия для перекрестных наводок и усложняет последующие этапы верификации. Раннее выявление таких участков позволяет перераспределять маршруты или инициировать корректировку размещения компонентов до завершения трассировки, избегая дорогостоящих итераций пересчета.

Алгоритм трассировки ячейки. Классический подход к трассировке предполагает итеративное построение недостающих проводников, и с этой точки зрения предлагаемый алгоритм не отличается от прочих [5].

Сложность на этом этапе представляет определение последовательности прокладки, поскольку порядок обработки напрямую влияет на итоговую плотность и качество топологии.

В текущей реализации приоритет задается на основе подсчета количества соединяемых групп точек для каждого потенциала. Чем больше групп требуется объединить, тем раньше выполняется трассировка соответствующего соединения. Такой подход позволяет сначала обрабатывать наиболее нагруженные цепи, снижая риск блокировки критических путей на поздних итерациях.

Ключевая идея состоит в том, чтобы минимизировать количество переходов и оптимизировать занимаемое пространство в процессе работы алгоритма, формируя группы контактов транзистора. При проектировании ячеек затвор транзистора допускает подключение с двух сторон, однако для корректной работы элемента достаточно соединения лишь с одной. Аналогичное правило применимо к активной области транзистора при достаточных геометрических размерах.

Поэтому точки подключения можно агрегировать в группы. В рамках такого подхода для установления связи с группой достаточно проложить проводник к любой ее точке. Этот же подход можно применить по отношению к контактным шинам, проходящим через всю ячейку.

Такая группировка контактов позволяет исключать необязательные межслойные переходы, что экономит площадь.

К тому же, поиском пути до группы, а не до каждой точки в отдельности, можно избавиться от необходимости выполнять трассировку до всех точек группы. Это достигается за счет введения дополнительного условия в работу алгоритма. Поиск пути прекращается в тот момент, когда найден путь хотя бы до одной точки группы при том, что стоимость этого пути заведомо меньше стоимости еще не найденных путей.

В качестве базы для модификаций был выбран волновой алгоритм на сетке [6]. Переход к оперированию группами вместо отдельных координат изменяет логику поиска пути. Схема алгоритма представлена на рис. 1.

Несмотря на групповую обработку, сохраняется проблема выбора конкретного маршрута при наличии нескольких вариантов с одинаковой стоимостью. В сеточном поиске пути такая ситуация возникает регулярно: несколько соседних ячеек могут иметь идентичную накопленную стоимость прокладки. Если в этот момент выбор осуществляется случайно или по жесткому приоритету направлений, проводник часто уходит в локально оптимальном, но глобально неэффективном направлении. Это приводит к образованию изолированных участков в трассировочном пространстве и блокировке путей для последующих соединений.

Для устранения этой неоднозначности предлагается заранее формировать карту напряженности для всей доступной сетки. Алгоритм ее построения опирается на вычисление кратчайшего расстояние от любой точки из группы до каждой клетки. Полученные значения суммируются по всем группам. В результате формируется поле, где низкие показатели соответствуют областям, находящимся в равной близости ко всем целевым группам, а высокие – удаленным или загроможденным участкам. Эта карта не зависит от текущего прокладываемого проводника и отражает общую топологическую «напряженность» в каждой клетке сетки трассировки.

При наличии равных по длине альтернатив построения пути, алгоритм обращается к карте напряженности как к дополнительному эвристическому критерию. Из равнозначных соседних ячеек выбирается та, которая имеет минимальное значение на карте напряженности. Таким образом, проводник естественным образом смещается в сторону центра масс всех ожидающих соединений, избегая тупиковых зон. Тот же механизм применяется для определения начальной точки трассировки. Вместо фиксированного порядка об-

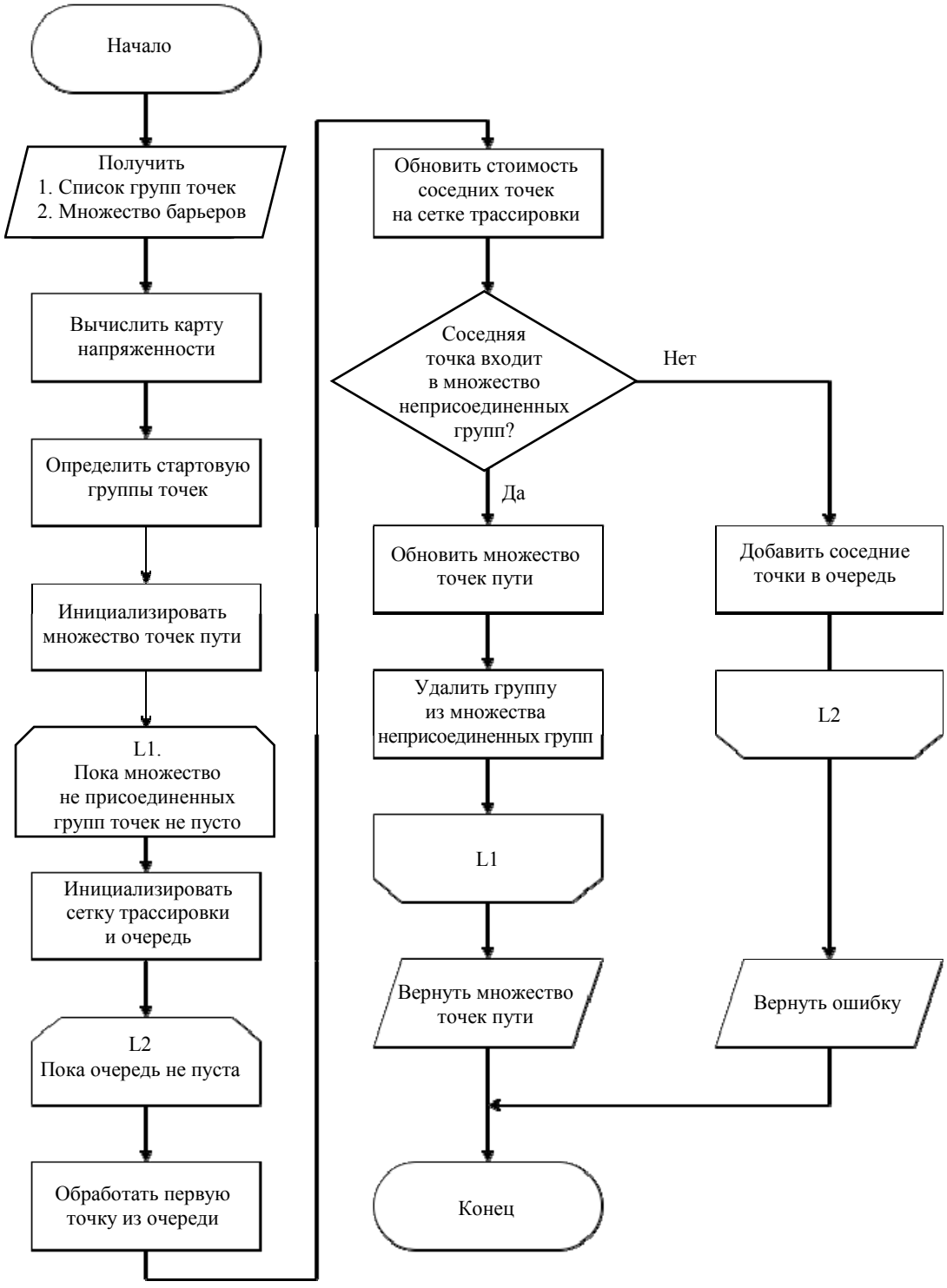


Рис. 1. Схема алгоритма трассировки
Fig. 1. The scheme of the routing algorithm

работки групп алгоритм анализирует сформированное поле напряженности и находит ячейку с абсолютным минимумом. Группа подключения, которой принадлежит эта точка, назначается стартовой. Это гарантирует, что процесс начинается из позиции, требующей наименьших первоначальных затрат на сближение с остальными элементами.

Важным аспектом практического применения разработанного алгоритма – обеспечение гибкости его конфигурации. Проектирование ячеек БИС требует учета разнообразных технологических норм и проектных сценариев, поэтому алгоритм должен допускать адаптацию к изменяющимся условиям без модификации базовой про-

граммной логики. С этой целью в архитектуру алгоритма интегрированы два ключевых инструмента управления: механизм резервирования трассировочных слоев [7] и матрица весовых коэффициентов, задающая условную стоимость прокладки в каждом допустимом направлении для каждого слоя.

Комбинация данных параметров обеспечивает тонкую калибровку поискового пространства, что позволяет применять единый алгоритмический каркас для решения широкого спектра задач. В качестве практического примера настройки можно рассмотреть управление использованием слоя поликремния. Повышение весовых коэффициентов для данного слоя искусственно увеличивает стоимость маршрутизации по нему, вследствие чего алгоритм будет применять поликремний исключительно в тех случаях, когда локальная прокладка оказывается экономичнее, чем формирование межслойного перехода на металлический уровень. Такой подход позволяет соблюдать технологические предпочтения без введения жестких запретов, сохраняя возможность принятия компромиссных решений при формировании топологии.

На рис. 2 приведен практический пример для трех групп точек. Каждая из групп точек представляет один элемент топологи. Группы 1 и 2

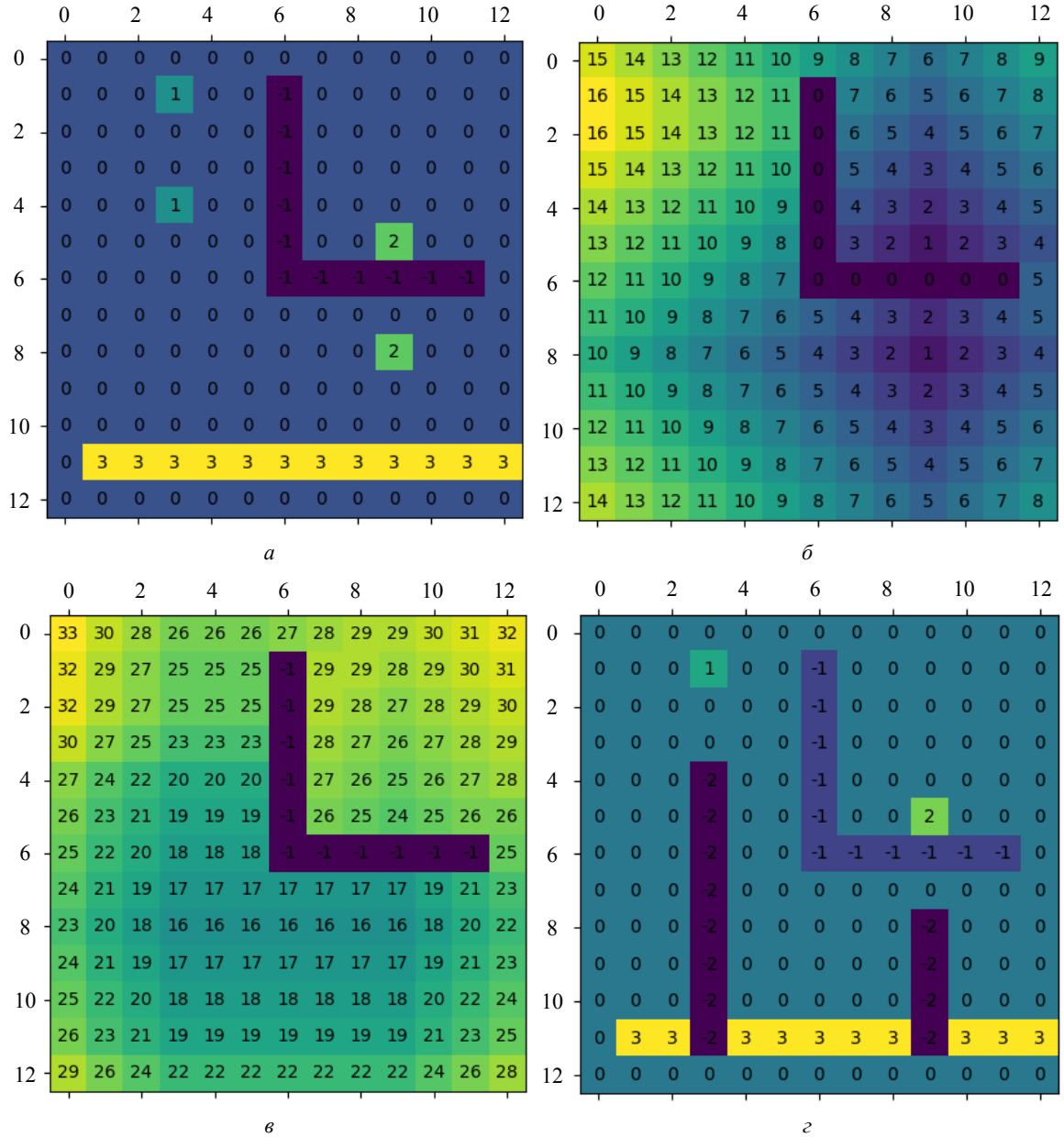


Рис. 2. Процесс трассировки трех групп точек: *а* – исходные группы точек; *б* – карта расстояний для группы 2; *в* – карта напряженности; *г* – результат трассировки
Fig. 2. The process of routing three groups of points: *а* – initial groups of points; *б* – distance map for group 2; *в* – tension map; *г* – the result of tracing

моделируют два транзистора в разных частях топологии. Группа 3 моделирует шину, к которой должны быть подключены транзисторы. После вычисления суммарных расстояний глобальный минимум фиксируется в ячейке с координатами (8, 9) (рис. 2, *б*). Эта позиция входит в состав группы 2, которая автоматически выбирается в качестве отправной. Далее алгоритм ищет путь к ближайшей целевой группе, на каждом шаге сверяя стоимость перехода со значениями карты напряженности. Маршрут прокладывается так, чтобы удерживать проводник в зонах с низкими суммарными расстояниями, что минимизирует риск пересечений и сокращает общую длину трассировки. Такой подход обеспечивает детерминированный выбор направления и снижает вероятность образования тупиковых конфигураций на ранних этапах итеративной прокладки.

Процесс построения пути на рис. 2 выполнялся для поля размером 13 × 13, координаты клеток трассировки обозначены по осям каждого поля. Визуализация процесса построения пути была выполнена реализацией представленного в данной статье алгоритма трассировки на языке Python с использованием библиотеки визуализации matplotlib. На рис. 2, *а* изображены исходные группы точек. На рис. 2, *б* представлен расчет карты расстояний от отправной точки – значения больше 0 означают расстояние до соответствующей клетки, значением 0 обозначены недоступные для трассировки клетки на сетке трассировки. На рис. 2, *в* представлена визуализация карты напряженности, где положительные значения в клетках означают оценку стоимости использования данной клетки в пути трассировки. На рис. 2, *г* выполнена визуализация результата трассировки, значением «–2» обозначен построенный путь, остальные значения соответствуют рис. 2, *а*.

Результаты. Описанный алгоритм был реализован в виде программы. Цель заключалась в проверке поведения алгоритма при введении жестких и мягких ограничений. В качестве входных данных использовались различные конфигурации алгоритма, представленные в табл. 1. Для каждого направления каждого слоя задается коэффициент стоимости одного шага. Всего использовалось три слоя для трассировки: поликремний (Poly), металл 1 (M1), металл 2 (M2).

В табл. 2 представлен полный набор полученных результатов работы алгоритма для ячеек с различными качественными характеристиками.

Каждая из рассмотренных конфигураций представляет собой трехмерную сетку трассировки, состоящую из 3 слоев: слоя поликремния (Poly) и двух слоев металла (M1, M2). Значения в табл. 1 означают стоимость перехода в соседнюю клетку в заданном направлении каждого из слоев. Направления обозначены буквами и соответствуют названиям осей трехмерного пространства *x*, *y*, *z* соответственно.

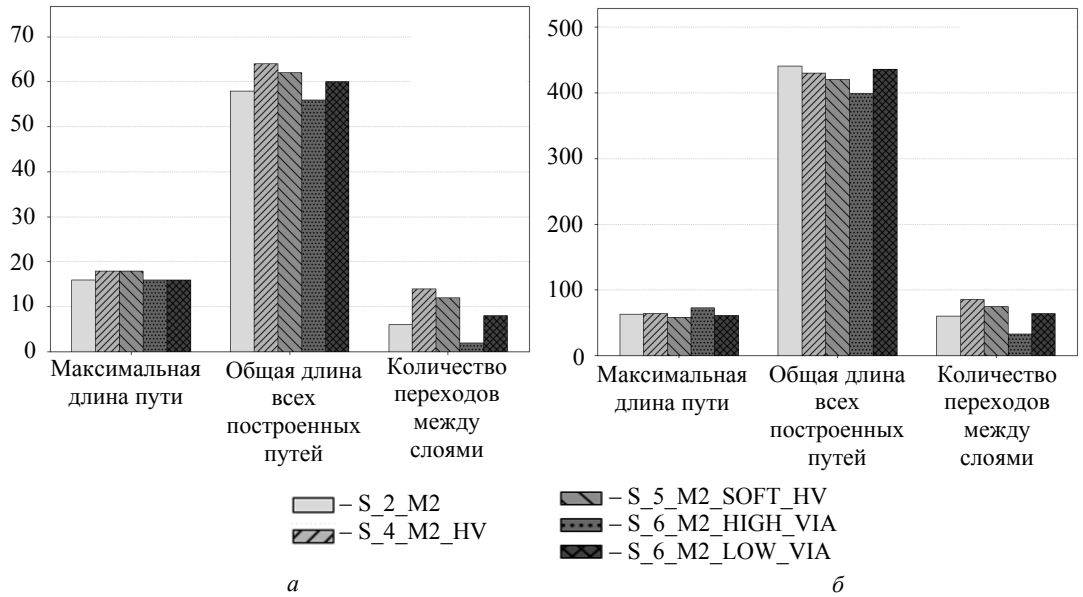
На рис. 3 представлены гистограммы результатов работы алгоритма трассировки для двух схем: сумматора из 28 транзисторов (sum.json), расположенных на двух парах диффузионных линеек, и мультиплексора из 6 транзисторов (mux.json) на одной паре диффузионных линеек. По оси абсцисс отложены три ключевых параметра качества трассировки: максимальная длина отдельного проводника, суммарная длина всех проводников и количество межслойных переходов. Шкала значений обозначает количество, для путей – количество клеток сетки трассировки.

Табл. 1. Тестовые конфигурации
Tab. 1. Test configurations

Название настройки	Описание	Стоимость перехода								
		Poly			M1			M2		
		X	*Y*	*Z*	*X*	*Y*	*Z*	*X*	*Y*	*Z*
S_2_M2	Базовый вариант настроек	5	5	5	1	1	5	1	1	5
S_6_M2_LOW_VIA	Сниженная стоимость переходов между слоями	5	5	2	1	1	2	1	1	2
S_6_M2_HIGH_VIA	Повышенная стоимость переходов между слоями	5	5	20	1	1	20	1	1	20
S_4_M2_HV	Жесткие ограничения направленности проводников для слоев M1, M2	5	5	5	1	–	5	–	1	5
S_5_M2_SOFT_HV	Мягкие ограничения направленности проводников для слоев M1, M2	5	5	5	1	5	5	5	1	5

Табл. 2. Результаты работы алгоритма
Tab. 2. The results of the algorithm's operation

Настройки	Время выполнения программы, с	Количество клеток поля	Суммарная длина проводников	Максимальная длина проводников	Количество переходов	Файл топологии
S_2_M2	0.517	675	58	16	6	mux.json
S_2_M2	4.540	1296	343	52	48	sum_1.json
S_2_M2	8.373	2028	441	63	60	sum_2.json
S_2_M2	1.845	864	203	40	32	cgb.json
S_6_M2_LOW_VIA	0.478	675	60	16	8	mux.json
S_6_M2_LOW_VIA	4.332	1296	350	55	53	sum_1.json
S_6_M2_LOW_VIA	7.638	2028	436	61	64	sum_2.json
S_6_M2_LOW_VIA	1.645	864	203	40	32	cgb.json
S_6_M2_HIGH_VIA	0.468	675	56	16	2	mux.json
S_6_M2_HIGH_VIA	4.485	1296	341	57	32	sum_1.json
S_6_M2_HIGH_VIA	7.848	2028	399	73	33	sum_2.json
S_6_M2_HIGH_VIA	1.998	864	209	44	27	cgb.json
S_4_M2_HV	0.412	675	64	18	14	mux.json
S_4_M2_HV	3.603	1296	342	51	63	sum_1.json
S_4_M2_HV	6.259	2028	430	64	86	sum_2.json
S_4_M2_HV	1.524	864	205	43	42	cgb.json
S_5_M2_SOFT_HV	0.590	675	62	18	12	mux.json
S_5_M2_SOFT_HV	5.006	1296	342	51	63	sum_1.json
S_5_M2_SOFT_HV	8.531	2028	421	58	75	sum_2.json
S_5_M2_SOFT_HV	2.146	864	201	41	40	cgb.json



а *б*

Рис. 3. Результаты трассировки при различных конфигурациях: *а* – для схемы мультиплексора; *б* – для схемы сумматора
Fig. 3. Routing results for different configurations: *а* – for the m ultiplexer circuit; *б* – for the adder circuit

Файлы топологии представлены в специализированном формате, содержащем описание элементов топологии и их координат.

Сравнение вариантов с жесткими и мягкими ограничениями направлений в слоях демонстрирует заметное преимущество гибкого подхода. При жесткой фиксации направлений алгоритм вынужден формировать обходные пути, что приводит к росту суммарной длины проводников и увеличению числа переходов, особенно в тополо-

гии сумматора. Внедрение мягких ограничений сохраняет общую направленность трассировки, но допускает локальные отклонения для поиска более коротких путей. В результате метрики улучшаются: суммарная длина сокращается, а количество переходов стабилизируется. Эффект становится более выраженным по мере увеличения площади топологии, что подтверждается данными для сумматора.

Управление стоимостью межслойных переходов напрямую влияет на баланс между компактностью и сложностью структуры. Повышение стоимости переходов закономерно снижает их количество, поскольку алгоритм отдает приоритет маршрутизации в пределах одного слоя, однако это достигается за счет удлинения критических путей. Максимальная длина проводника в большинстве случаев возрастает, суммарная длина при этом ведет себя неоднозначно. В простых схемах – таких, как мультиплексор, снижение числа переходов может компенсироваться удлинением трасс, тогда как в сложных структурах общая длина нередко уменьшается за счет более рационального использования доступных слоев. Обратная ситуация наблюдается при низкой стоимости переходов, где алгоритм активно использует межслойные соединения для минимизации длины, что увеличивает их количество.

Время выполнения алгоритма зависит от количества проводников, размера трассируемой области и качества предварительного размещения компонентов. На практике это проявляется в существенном разбросе вычислительных затрат даже для схожих по площади топологий. Например, при трассировке одной и той же электрической схемы время расчета может различаться даже в два раза исключительно из-за различий в плотности и равномерности размещения элементов. Полученные данные подтверждают, что универсальных настроек не существует. Параметры ограничений и весовых коэффициентов необходимо подбирать под конкретную задачу, последовательно варьируя их для достижения оптимального баланса между задержками, площадью и технологичностью изготовления.

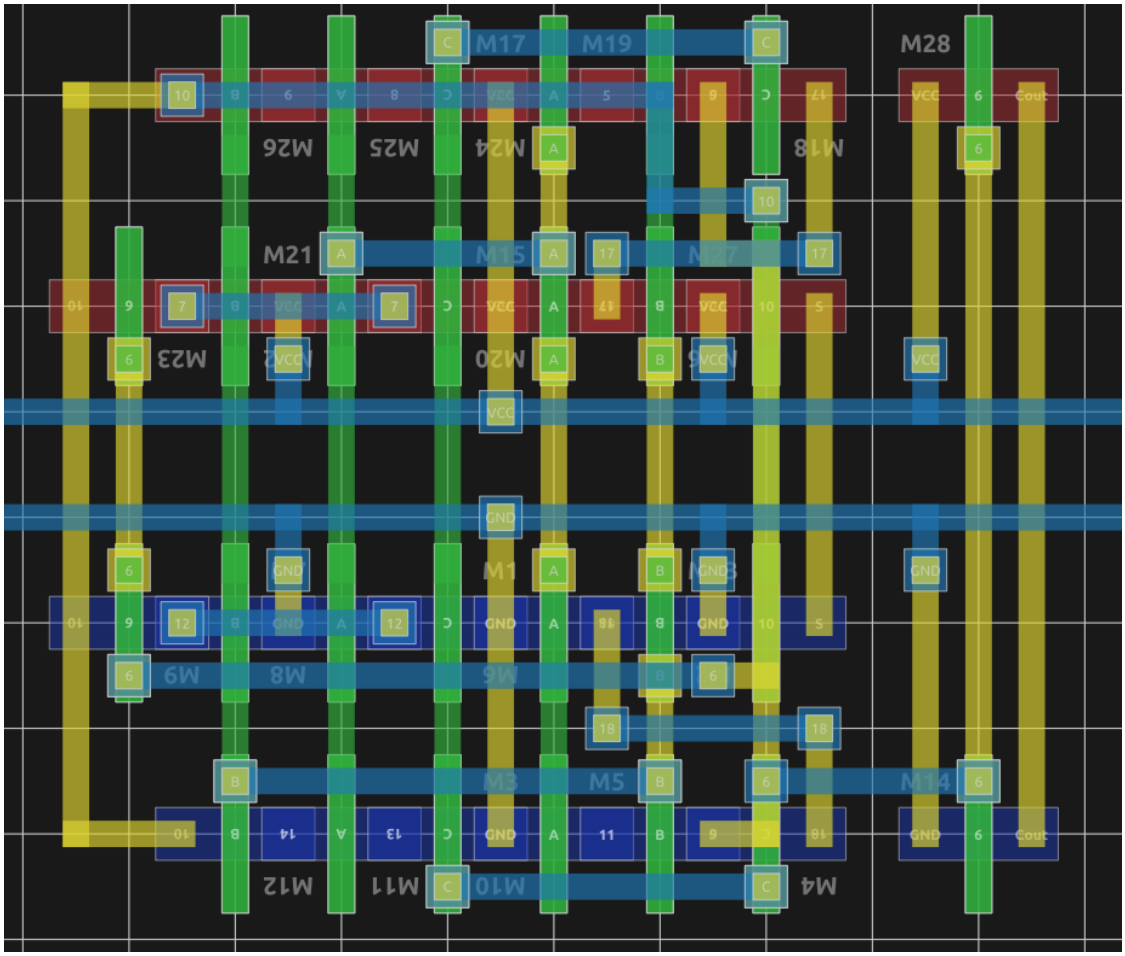


Рис. 4. Инвариантная топология сумматора после трассировки
Fig. 4. Invariant adder layout after tracing

На рис. 4 представлена визуализация результата работы алгоритма трассировки для схемы сумматора.

Данный рисунок демонстрирует результат применения представленного алгоритма и показывает, что для одной из типичных схем не происходит каких-то фатальных ошибок при построении путей.

Заключение. В ходе работы удалось реализовать алгоритм трассировки, демонстрирующий достаточное качество и стабильную эффективность на типовых тестовых примерах. Отличительной особенностью предложенного метода служит подход на основе «групп», который позволяет структурировать соединения и последовательно удовлетворять проектные ограничения. Такая организация обеспечивает баланс между вычислительной сложностью и точностью формирования проводников.

Экспериментальная проверка подтвердила ряд закономерностей. Введение мягких ограничений на направления трассировки в слоях дает заметное улучшение характеристик по сравнению с жесткими правилами, причем эффект усиливается с ростом размеров топологии. Повышение стоимости переходов между слоями закономерно сокращает их количество, однако при этом, как правило, увеличивается длина наиболее протяженного проводника. Общая длина трассировки может как расти, так и уменьшаться в зависимости от структуры схемы, что подчеркивает необходимость индивидуального подбора параметров для каждой конкретной задачи.

Время выполнения алгоритма определяется количеством проводников, размерами трассируемой области и качеством предварительного размещения компонентов. На практике это проявляется в существенном разбросе вычислительных затрат: при идентичной электрической схеме и сопоставимой площади топологии время расчета может отличаться в два раза исключительно из-за различий в качестве размещения. При этом реализованная архитектура алгоритма остается гибкой. Возможность оперативной корректировки ограничений и стоимостных параметров делает предложенный метод универсальным инструментом для инвариантного проектирования.

Список литературы

1. Sherwani N. A. Algorithms for VLSI physical design automation. New York: Springer, 2005. 568 p.

2. Weste N. H. E. CMOS VLSI Design: A circuits and systems perspective. Boston: Addison-Wesley, 2011. 1024 p.

3. Sait S. M., Habib Y. VLSI physical design automation: Theory and practice. Singapore: World Scientific, 1999. 456 p.

4. Pan M., Chu C. FastRoute 2.0: A High-quality and efficient global router // 2007 Asia and South Pacific Design Automation Conf. Yokohama, Japan: IEEE, 2007. P. 250–255. doi: 10.1109/ASPDAC.2007.357994.

5. Global routing in VLSI: Algorithms, theory, and computation practice / A. Deza, C. Dickson, T. Terlaky, A. Vannelli, Hu. Zhang // J. of Combinatorial Math. and Combinatorial Comp. (JCMCC). 2011. Vol. 80. P. 1–25.

6. Lee C. Y. An algorithm for path connections and its applications // IRE Trans. on Electron. Comp. 1961. Vol. EC-10, no. 3. P. 346–365.

7. Shiryaev P. M., Mironov S. E. Optimization of IC cell routing in conditions of uncertainty // 2025 XXVIII Intern. Conf. on Soft Comp. and Measurements (SCM). Saint Petersburg, Russia. 2025. P. 123–126. doi: 10.1109/SCM66446.2025.11060254.

Информация об авторе

Ширяев Павел Михайлович – аспирант кафедры вычислительной техники СПбГЭТУ «ЛЭТИ».
E-mail: pshir2000z@gmail.com

References

1. Sherwani N. A. Algorithms for VLSI physical design automation. New York: Springer, 2005. 568 p.

2. Weste N. H. E. CMOS VLSI Design: A circuits and systems perspective. Boston: Addison-Wesley, 2011. 1024 p.

3. Sait S. M., Habib Y. VLSI physical design automation: Theory and practice. Singapore: World Scientific, 1999. 456 p.

4. Pan M., Chu C. FastRoute 2.0: A High-quality and efficient global router // 2007 Asia and South Pacific Design Automation Conf. Yokohama, Japan: IEEE, 2007. P. 250–255. doi: 10.1109/ASPDAC.2007.357994.

5. Global routing in VLSI: Algorithms, theory, and computation practice / A. Deza, C. Dickson, T. Terlaky, A. Vannelli, Hu. Zhang // J. of Combinatorial Math. and Combinatorial Comp. (JCMCC). 2011. Vol. 80. P. 1–25.

6. Lee C. Y. An algorithm for path connections and its applications // IRE Trans. on Electron. Comp. 1961. Vol. EC-10, no. 3. P. 346–365.

7. Shiryaev P. M., Mironov S. E. Optimization of IC cell routing in conditions of uncertainty // 2025 XXVIII Intern. Conf. on Soft Comp. and Measurements (SCM). Saint Petersburg, Russia. 2025. P. 123–126. doi: 10.1109/SCM66446.2025.11060254.

Information about the author

Pavel M. Shiryaev – postgraduate student of the Department of Computer Engineering of Saint-Petersburg Electrotechnical University.
E-mail: pshir2000z@gmail.com

Статья поступила в редакцию 21.04.2026; принята к публикации после рецензирования 19.09.2026; опубликована онлайн 27.11.2026.

Submitted 21.04.2026; accepted 19.09.2026; published online 27.11.2026.