

# ОПТИЧЕСКИЕ ВЫЧИСЛИТЕЛИ НЕЛИНЕЙНЫХ ФУНКЦИЙ

**Каменский Владислав Валерьевич,**  
Ростовский государственный университет путей  
сообщения", г. Ростов-на-Дону, Россия,  
[kam-vladislav@yandex.ru](mailto:kam-vladislav@yandex.ru)

DOI: 10.36724/2072-8735-2026-20-3-35-42

**Manuscript received** 20 December 2025;  
**Accepted** 26 February 2026

**Соколов Сергей Викторович,**  
Московский технический университет связи и  
информатики, Москва, Россия,  
[s.v.s.888@yandex.ru](mailto:s.v.s.888@yandex.ru)

**Лагунова Елена Олеговна,**  
Ростовский государственный университет путей  
сообщения", г. Ростов-на-Дону, Россия,  
[lagunova@rambler.ru](mailto:lagunova@rambler.ru)

**Ключевые слова:** оптическое вычислительное  
устройство, извлечение квадратного корня, возведение в  
степень, гигагерцовый диапазон, оптический  
разветвитель, транспарант, оптический объединитель

В работе рассмотрен новый подход к конструированию средств высокопроизводительной вычислительной техники на основе оптоэлектронных технологий. Оптические устройства обработки информации применяются в областях, где требуется максимальное быстродействие и простота вычислений, например, при синтезе встраиваемых систем и специализированных вычислителей для обработки высокоскоростных процессов. В общем случае встраиваемые системы обрабатывают поступающие данные вполне определенного и заранее известного диапазона, при этом вычисления допускается выполнять с заданной точностью, но время вычисления весьма ограничено и критично. В связи с этим, для разрабатываемого оптического вычислителя, в результате проведения анализа существующих форматов представления чисел, был выбран формат представления чисел с фиксированной запятой. В качестве примера реализуемой нелинейной функции была выбрана одна из наиболее сложных функций для вычисления в традиционных схемах - функция извлечения квадратного корня. Проведен анализ существующих алгоритмов извлечения квадратного корня как базовой операции многих математических вычислений, находящихся свое практическое применение в геометрии, архитектуре, строительстве и пр. Изложены теоретические основы построения быстродействующего вычислителя квадратного корня для чисел, представленных в двоичном коде. Предложенный подход позволяет производить вычисление квадратного корня на основе комбинационного устройства, без использования метода проб и ошибок, что радикально увеличивает скорость вычислений. Так как при решении практических задач необходимо за минимальное время производить целый набор вычислений над несколькими аргументами, то с этой целью разработана функциональная схема специализированного оптического программируемого устройства для выполнения набора функциональных преобразований на основе принципов построения и оптимизации комбинационных схем. Основными базовыми элементами оптического программируемого устройства являются оптические разветвители, объединители и транспаранты, технология изготовления которых известна и не вызывает трудностей. Программирование оптического вычислителя для реализации конкретной функции осуществляется разрушением оптических связей, не участвующих в формировании логических функций. Минимальное количество элементов и высокая скорость вычислений делают оптическое программируемое устройство перспективным при построении быстродействующих технических средств вычислительной техники и систем связи.

#### Информация об авторах:

**Каменский Владислав Валерьевич**, доцент, к.т.н., ФГБОУ ВО "Ростовский государственный университет путей сообщения", Ростов-на-Дону, Россия, ORCID: 0000-0002-0704-8686

**Соколов Сергей Викторович**, профессор, д.т.н., ФГБОУ ВО "Московский технический университет связи и информатики", Москва, Россия, ORCID: 0000-0002-5246-841X

**Лагунова Елена Олеговна**, профессор, д.т.н., ФГБОУ ВО "Ростовский государственный университет путей сообщения", г. Ростов-на-Дону, Россия, ORCID: 0000-0002-2761-8068

#### Для цитирования:

Каменский В.В., Соколов С.В., Лагунова Е.О. Оптические вычислители нелинейных функций // Т-Comm: Телекоммуникации и транспорт. 2026. Том 20. №3. С. 35-42.

#### For citation:

V.V. Kamenskij, S.V. Sokolov, E.O. Lagunova, "Fast optical computers," T-Comm, 2026, vol. 20, no.3, pp. 35-42. (in Russian)

## Введение

Объемы информационных потоков, генерируемых, передаваемых и потребляемых человечеством, неуклонно растут с каждым годом, что требует все возрастающих вычислительных мощностей. Дальнейшее увеличение вычислительных возможностей при обработке информации традиционными (электронными устройствами) невозможно. Это обусловлено невозможностью преодолеть термодинамический, квантово-механический предел и предел скорости распространения сигналов. Что, в свою очередь, приводит к поиску новых подходов при построении вычислительных устройств, базирующихся на принципиально новых идеях и концепциях. В частности, оптических систем, не обладающих недостатками электронных устройств и имеющих высокое быстродействие при возможности микроминиатюризации и минимальном энергопотреблении.

Вычисление нелинейных функций является сложной задачей, решение которой основывается на аналитических или итерационных методах. Одной из таких задач является извлечение корня.

Наибольшее применение операция извлечения корня получила в геометрии и связанных с ней областях практического применения в архитектуре и строительстве.

Методы вычисления корней (квадратных, кубических и высших) включают итеративные алгоритмы, такие как метод Ньютона-Рафсона, бинарный поиск и специализированные схемы для многочленов. Современные подходы фокусируются на ускорении сходимости, аппаратной реализации и обработке множественных корней.

Метод Ньютона-Рафсона остается основным для приближенного вычисления корней уравнений  $f(x)=0$ , обеспечивая квадратичную сходимость при хорошем начальном приближении [1].

Вавилонский (Геронский) метод специально для квадратных корней использует итерацию  $x_{n+1} = \left(x_n + \frac{a}{x_n}\right) / 2$ , показывая быструю сходимость для положительных  $a$ . Эти методы модифицированы для численной устойчивости в многомерных задачах [2].

Для корней многочленов применяют схему Горнера для оценки, метод Лобачевского для интервалов и итеративное разложение на множители, эффективно выявляя комплексные и кратные корни. Новые алгоритмы сочетают эти подходы с машинным обучением для предсказания интервалов. Аппаратные реализации на ПЛИС ускоряют вычисления сигмоидов и корней в реальном времени [3, 4].

Авторы [5] исследуют эффективность различных методов извлечения корней и делают сравнительный анализ преимуществ и недостатков каждого метода. В работе [6] представлены новейшие разработки и идеи по извлечению корней, демонстрируется использование графических моделей и компьютеров для оптимизации решений. В [7] изложено развитие методов извлечения корней с демонстрацией современных достижений математики в этой сфере.

В двоичной системе описаны схемы для квадратных и кубических корней, аналогично делению столбиком [8-17].

Операция извлечения квадратного корня в двоичном формате может быть решена путем последовательного подбора

цифр результата, с использованием регистров и сумматоров. Метод реализуется путем применения сдвигов и вычитаний. Вторым способом является табличный способ. Заранее рассчитываются результаты вычислений и заносятся в таблицу. При необходимости вычислить корень, в таблице находится нужное значение «число в степени», а результат основание степени выбирается из таблицы.

## Материалы и методы

Методы построения описываемых в данной статье оптических вычислителей основаны на оптических технологиях. Оптические технологии – это направление науки и техники, которое использует оптические сигналы для реализации ключевых информационных процессов. Преимуществами оптических технологий является сверхвысокая скорость передачи информации, минимальная мощность передаваемого сигнала и минимальные потери сигнала, позволяющие разрабатывать быстродействующие оптические устройства со сверхнизким потреблением энергии.

В настоящий момент оптические технологии получили распространение в устройствах телекоммуникации и связи, позволяя передавать информацию на дальние расстояния с малыми задержками. В оптических приемо-передающих системах передача информации осуществляется практически без потерь, обусловленных электромагнитными помехами, обеспечивая высокоскоростной обмен данными. Еще одним важным свойством оптических систем является возможность реализации параллельных вычислений.

Сегодняшние решения – микролинзы, дифракционные решетки, фотонные кристаллы, интегральные оптические схемы и метаматериалы – отличаются микронными габаритами и прецизионной точностью, открывая путь к разработке компактных и высокопроизводительных информационных устройств.

Источниками оптического излучения могут быть лампы, светодиоды и лазеры. Спектральный диапазон таких источников охватывает интервал от 10 нм до 1 мм. Также существуют источники оптического излучения на основе фотонных кристаллов, квантовых точек, суперлюминесцентных диодов и др.

Во всех оптических устройствах традиционно используются три вида базовых оптических элементов, составляющих его основу [18]. Это оптические волноводы, оптические разветвители и оптические объединители.

Перечисленные оптические компоненты – волноводы, разветвители и объединители – выполняют исключительно функции передачи, распределения и объединения оптических сигналов. Они не обладают способностью реализовывать логические операции (И, ИЛИ, НЕ и др.). Такими устройствами могут быть трансфазоры, оптически связанные волноводы. Интенсивность сигналов на выходе этих элементов скачкообразно изменяется при плавном изменении сигнала на входе. Схема и подробное функционирование оптических логических элементов описаны в [19-21].

## Выбор формата данных

В электронных вычислительных машинах целые числа  $X$ , состоящие из  $N$  разрядов, хранятся в виде двоичных чисел:

$$X = 2^{N-1}x_{N-1} + \dots + 2^M x_M + \dots + 2^1 x_1 + 2^0 x_0,$$

что позволяет хранить очень большие числа. Несмотря на это, такой тип хранения данных имеет существенный недостаток: отсутствует возможность хранения дробных значений.

Так как при вычислении квадратного корня могут быть получены дробные значения, то целочисленные типы чисел не подходят для операций вычисления корня.

Для этих целей используют числа с плавающей запятой:

$$X = a \cdot q^N$$

состоящие из  $a$ -мантисы и  $N$ -порядка.

Операции с целыми числами производятся быстро, а операции с числами с плавающей запятой выполняются значительно дольше и требуют специальных аппаратных устройств. Поэтому во встраиваемых системах, обрабатывающих данные вполне определенного и заранее известного диапазона, используются числа с фиксированной запятой. Числа с фиксированной запятой содержат, как целую  $I$ , так и дробную часть  $F$ :

$$X = I + F,$$

где

$$I = 2^{n-1}x_{n-1} + 2^i x_i + \dots + 2^1 x_1 + 2^0 x_0,$$

$$F = 2^{1/2} x_{f-1} + 2^{1/4} x_{f-2} \dots + 2^{1/f} x_0$$

В результате число с фиксированной запятой имеет вид

$$X = 2^{n-1}x_{n-1} + \dots + 2^1 x_{m+1} + 2^0 x_m + 2^{-1} x_{m-1} + 2^{-2} x_{m-2} \dots + 2^{-f} x_0$$

Положение запятой определяется количеством бит как целой  $I$ , так и дробной частей  $F$ . В приведенном примере фиксированная запятая располагается между разрядами  $x_m$  и  $x_{m-1}$

Для обозначения положения фиксированной запятой используют количество бит в целой части и количество бит в дробной части  $Q_i.f$ . Количество бит в целой и дробной части может совпадать (Q4.4, Q8.8, Q16.16), а может и отличаться (Q8.16, Q3.5). Например, десятичному числу 1.4375 в формате числа с фиксированной запятой Q4.4 будет соответствовать двоичное число 0001.0111.

### Оптическое комбинационное вычислительное устройство

Рассмотрим далее описание структуры и принципа действия оптического вычислителя.

При небольшом диапазоне значений  $X$  вычисление корня может осуществляться табличным методом.

Устройство, реализующее табличный метод, состоит из дешифратора, ячеек памяти и схемы коммутации выходных сигналов. Дешифратор осуществляет выбор одной ячейки памяти со значением результата. В ячейку памяти заранее записывается значение вычисленного корня. Схема коммутации выходных сигналов позволяет передать на выход устройства сигналы только одной ячейки памяти.

Другим, более простым, способом получения результата операции извлечения квадратного корня является построение схемы вычислительного устройства как комбинационной схемы.

Это возможно благодаря тому, что состояние каждого выхода вычислительного устройства однозначно определяется входными сигналами:

$$Y_i = f(x_{n-1}, \dots, x_i, \dots, x_1, x_0)$$

Запишем, например, для входных сигналов  $x_2, x_1, x_0$  значения выходных сигналов  $y_7, y_6, y_5, y_4, y_3, y_2, y_1, y_0$  в формате с фиксированной запятой Q4.4 и в двоичном виде  $M = \{0000.0000, 0001.0000, 0001.0111, 0001.1100, 0010.0000, 0010.0100, 0010.0111, 0010.1010\}$ .

Логические функции для выходных сигналов  $y_7, y_6, y_5, y_4$ , определяющих целую часть, в зависимости от входных сигналов  $x_2, x_1, x_0$  представлены ниже

$$y_7 = 0$$

$$y_6 = 0$$

$$y_5 = x_2 \bar{x}_1 \bar{x}_0 \vee x_2 \bar{x}_1 x_0 \vee x_2 x_1 \bar{x}_0 \quad x_2 \bar{x}_1 x_0 = x_2$$

$$y_4 = \bar{x}_2 \bar{x}_1 x_0 \vee \bar{x}_2 x_1 \bar{x}_0 \vee \bar{x}_2 x_1 x_0$$

Дробную часть результата определяют выходные сигналы  $y_3, y_2, y_1, y_0$ . Логические функции для выходных сигналов  $y_3, y_2, y_1, y_0$  равны

$$y_3 = \bar{x}_2 x_1 x_0 \vee x_2 x_1 x_0$$

$$y_2 = \bar{x}_2 x_1 \bar{x}_0 \vee \bar{x}_2 x_1 x_0 \vee x_2 \bar{x}_1 x_0 \vee x_2 x_1 \bar{x}_0$$

$$y_1 = \bar{x}_2 x_1 \bar{x}_0 \vee x_2 x_1 \bar{x}_0 \vee x_2 x_1 x_0$$

$$y_0 = \bar{x}_2 x_1 \bar{x}_0 \vee x_2 x_1 \bar{x}_0$$

Запишем логические функции для выходных сигналов  $y_5, y_4, y_3, y_2, y_1, y_0$  в виде чисел на входе, при котором функция принимает состояние единицы:

$$y_5 = 4, 5, 6, 7$$

$$y_4 = 1, 2, 3$$

$$y_3 = 3, 7$$

$$y_2 = 2, 3, 5, 6$$

$$y_1 = 2, 6, 7$$

$$y_0 = 2, 6$$

Отметим, что значения логической функции  $y_5$  полностью соответствует входному сигналу  $x_2$ , и результат упрощения это подтверждает:

$$y_5 = x_2 \bar{x}_1 \bar{x}_0 \vee x_2 \bar{x}_1 x_0 \vee x_2 x_1 \bar{x}_0 \vee x_2 x_1 x_0 =$$

$$= x_2 (\bar{x}_1 \bar{x}_0 \vee \bar{x}_1 x_0 \vee x_1 \bar{x}_0 \vee x_1 x_0) = x_2 \cdot 1 = x_2$$

В общем виде оптическое устройство извлечения квадратного корня строится на основе оптических инверторов  $I_0 \dots I_{i-1}$ , оптических конъюнкторов  $F_0 \dots F_{i-1}$  и оптических дизъюнкторов  $V_0 \dots V_{i-1}$  и работает следующим образом.

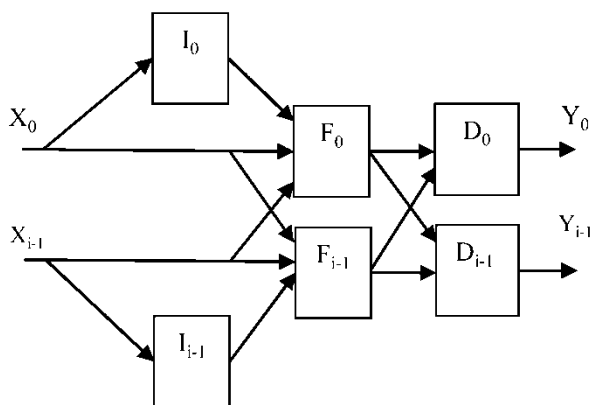


Рис. 1. Оптическое комбинационное вычислительное устройство

Входные оптические сигналы  $x_{i-1} \dots x_0$  подаются на входы оптических конъюнкторов  $F_0 \dots F_{i-1}$  и при необходимости – на входы оптических инверторов  $I_i$ . Оптические конъюнкторы  $F_0 \dots F_{i-1}$  осуществляют операции логического умножения «И» в соответствии с логическими функциями для выходных сигналов  $y_7, y_6, y_5, y_4, y_3, y_2, y_1, y_0$ .

Количество входов каждого оптического конъюнктора  $F_0 \dots F_{i-1}$  может быть от 1 до  $m$ . Для формирования логической функции одного выхода  $y_i$  понадобится от 1 до  $2^m$  оптических конъюнкторов  $F_0 \dots F_{i-1}$ . С выходов оптических конъюнкторов  $F_0 \dots F_{i-1}$  оптические сигналы  $P_m \dots P_0$  подаются на входы оптических дизъюнкторов  $B_0 \dots B_{i-1}$ . Если результат конъюнкции  $P_i$  используется при формировании выходной функции  $Y_i$ , то выход оптического конъюнктора  $F_i$  подключается ко входу соответствующего оптического дизъюнктора  $B_i$ .

Таким образом, при подаче на вход устройства оптических сигналов  $x_{i-1} \dots x_0$ , на выходе устройства будут появляться оптические сигналы  $y_7, y_6, y_5, y_4, y_3, y_2, y_1, y_0$ , соответствующие результату вычисления квадратного корня числа, поданного на вход устройства.

### Оптическое программируемое вычислительное устройство

При решении практических задач необходимо за минимальное время вычислять не одну единственную функцию, а произвести целый набор вычислений над несколькими аргументами.

Например, сложная функция может иметь вид:

$$y = f(h(g(a), g(b)))$$

В первую очередь вычисляются значения функций  $ta=g(a)$  и  $tb=g(b)$ .

Результаты вычисления  $ta$  и  $tb$  становятся аргументами для функции  $q=h(ta, tb)$  и в последнюю очередь вычисляется функция  $y=f(q)$ .

Допустим, требуется выполнить несколько действий над двумя числами  $a$  и  $b$ .

$$c = \sqrt{a^2 + b^2}$$

Традиционный подход заключается в следующем:

1. Разрабатывается устройство, обеспечивающее возведение числа в квадрат.

2. Разрабатывается устройство суммирования результата возведения первого и второго числа в квадрат (в традиционных способах суммирование выполняется с использованием переносов, что приводит к значительным временным затратам).

3. Разрабатывается устройство извлечения квадратного корня.

4. Разрабатывается общее устройство на основе ранее разработанных элементов.

Таким образом, результат вычисляется в виде нескольких последовательных операций. Причем, каждое следующее действие осуществляется только после полного завершения расчета предыдущего значения. Устройство, реализующее каждое действие, разрабатывается и оптимизируется независимо от остальных. В результате, у таких устройств наблюдается большое время вычисления результата за счет последовательного вычисления и переизбытка логических элементов по причине отсутствия оптимизации между ними.

Рассмотрим другой подход вычисления сложной функции. Подход основан, во-первых, на проектировании устройства вычисления конкретной сложной функции, а не взаимосвязанного набора различных специализированных устройств, а во-вторых, на использовании результата, а не вычислении значения путем пробных значений.

Введем некоторые ограничения. Числа, подаваемые на вход,  $a$  и  $b$  – целые. Требуется получить результат в формате с фиксированной запятой (точкой) Q4.4.

Например, для входных сигналов  $a_1, a_0, b_1, b_0$ , которые могут принимать значения  $\{0000, 0001, \dots, 1111\}$  вычислим значения выходных сигналов в десятичном виде  $D = \{0.0000, 1.0000, 2.0000, 3.0000, 1.0000, 1.4142, 2.2361, 3.1623, 2.0000, 2.2361, 2.8284, 3.6056, 3.0000, 3.1623, 3.6056, 4.2426\}$ . Затем переведем их в двоичный формат с фиксированной запятой Q4.4  $M[] = \{0000.0000, 0001.0000, 0010.0000, 0011.0000, 0001.0000, 0001.0111, 0010.0100, 0011.0011, 0010.0000, 0010.0100, 0010.1110, 0011.1010, 0011.0000, 0011.0011, 0011.1010, 0100.0100\}$ .

Для каждого разряда целой части запишем логические функции для выходных сигналов  $c_7, c_6, c_5, c$ , определяющих целую часть в зависимости от входных сигналов  $a_1, a_0, b_1, b_0$

$$c_7 = 0$$

$$c_6 = a_1 a_0 b_1 b_0$$

$$c_5 = \bar{a}_1 b_1 \vee a_1 \bar{b}_1 \vee \bar{a}_0 b_1 \vee b_1 \bar{a}_0$$

$$c_4 = \bar{a}_1 b_0 \vee a_0 \bar{b}_1 \vee \bar{a}_0 b_1 b_0 \vee a_1 a_0 \bar{b}_0$$

Логические функции для выходных сигналов дробной части результата  $c_3, c_2, c_1, c_0$  представлены ниже

$$c_3 = a_1 \bar{a}_0 b_1 \vee a_1 b_1 \bar{b}_0$$

$$c_2 = \bar{a}_1 a_0 \bar{b}_1 b_0 \vee \bar{a}_1 a_0 b_1 \bar{b}_0 \vee a_1 \bar{a}_0 \bar{b}_1 b_0 \vee a_1 \bar{a}_0 b_1 \bar{b}_0 \vee a_1 a_0 b_1 b_0$$

$$c_1 = \bar{a}_1 a_0 b_0 \vee a_1 \bar{a}_0 \bar{b}_1 \vee a_0 \bar{b}_1 b_0 \vee a_1 b_1 \bar{b}_0$$

$$c_0 = \bar{a}_1 a_0 b_0 \vee a_0 \bar{b}_1 b_0$$

**Функциональная схема оптического вычислителя**

Так, функциональная схема оптического вычислителя, осуществляющего вычисление трех арифметических функций, состоит из:

- 1) оптических инверторов  $I_i, i=0,1,2,\dots,m$ ;
- 2) оптических конъюнкторов  $F_i, i=0,1,2,\dots,m$ ;
- 3) оптических транспарантов  $B_i, i=0,1,2,\dots,m$ ;
- 4) оптических разветвителей  $S_i, i=0,1,2,\dots,m$ ;
- 5) оптических дизъюнкторов  $D_i, i=0,1,2,\dots,m$ ;

Входные сигналы  $X_0 \dots X_m$  поступают на входы оптических инверторов  $I_i$  и оптических конъюнкторов  $F_i$ . Выходы оптических конъюнкторов  $F_i$  подключены к входам оптических транспарантов  $B_i$ . Выходы оптических транспарантов  $B_i$  через оптические разветвители подключены ко входам оптических дизъюнкторов  $D_i$ .

Особенности оптических конъюнкторов  $B_i$  в том, что единичный сигнал на выходе составляет  $K$  усл. ед.

Оптические транспаранты имеют такой коэффициент поглощения, что оптические сигналы на выходах каждого оптического разветвителя  $S_i$  равны 1 усл. ед.

Особенности оптических дизъюнкторов  $D_i$  состоят в том, что единичный сигнал на его входе интенсивностью больше 0,5 усл. ед. приводит к появлению сигнала на выходе 1 усл. ед.

Оптические транспаранты  $B_i$ , оптические разветвители  $S_i$  и оптические дизъюнкторы  $D_i$  составляют оптическую логическую матрицу.

Для реализации требуемых логических зависимостей между входами и выходами устройства осуществляется разрушение оптических связей в области  $O_1$  между входами устройства или выходами оптических инверторов и входами оптических конъюнкторов  $F_i$ , а также разрушение связей в области  $O_2$  между выходами оптических разветвителей  $S_i$  и входами оптических дизъюнкторов  $D_i$ .

**Принцип действия оптического вычислителя**

На входы оптического устройства, осуществляющего вычисление трех арифметических функций, подаются числа  $A$  и  $B$ . Из чисел  $A$  и  $B$  формируется число двойной разрядности  $X$ , так, что старшая часть числа  $XH$  соответствует числу  $A$ , а младшая часть числа  $XL$  соответствует числу  $B$ .

Полученное число  $X$  подается на вход устройства в двоичном позиционном коде " $x_m, x_{m-1}, \dots, x_0$ " в виде набора из  $m$  бит  $x_i$ :

$$X = \sum_{i=0}^{m-1} x_i 2^i$$

Входные оптические сигналы с интенсивностью 1 усл.(овных).ед.(иниц) поступают на входы оптических конъюнкторов  $F_i$ .

Рассмотрим формирование выходных сигналов для каждого выхода  $c_7, c_6, c_5, c$ , определяющего целую часть.

Сигнал  $c_7$  при указанном выше наборе входных сигналов всегда равен нулю. Поэтому разрушаются все связи между выходами оптических разветвителей  $S_i$  и входами оптического дизъюнктора  $D_7$ .

Для формирования сигнала  $c_6$  используется оптический конъюнктор  $F_0$ . На входы конъюнктора  $F_0$  подаются четыре сигнала  $a_1, a_0, b_1, b_0$ . Сигнал  $c_6$  принимает значение единицы, если на всех входах устройства присутствуют единицы. При

наличии на входах оптического конъюнктора  $F_0$  комбинации сигналов

$$a_1, a_0, b_1, b_0 = 1, 1, 1, 1$$

оптический конъюнктор  $F_0$  выдает оптический сигнал в  $K$  условных единиц на свой выход. Пройдя оптический разветвитель  $S_i$ , оптический сигнал уменьшится и с интенсивностью 1 усл. ед. поступит на вход оптического дизъюнктора  $D_6$ . В данном случае кроме конъюнктора  $F_0$  больше ни один конъюнктор  $F_i$  не выдает сигнал высокого уровня интенсивности  $K$  усл. ед. на свой выход, остальные сигналы на выходах конъюнкторов  $F_i$  равны 0. Так как интенсивность сигнала на входе дизъюнктора  $D_6$  больше 0,5, то интенсивность сигнала на его выходе будет равна 1 усл. ед. В результате интенсивность сигнала на входе дизъюнктора  $D_6$  будет равна 1.

$$c_6 = D_6(F_0(a_1 a_0 b_1 b_0))$$

Для формирования сигнала  $c_5$  используется 4 конъюнктора  $F_1 \dots F_4$ . На входы конъюнкторов  $F_1 \dots F_4$  входные сигналы подаются парами. На входы конъюнктора  $F_1$  подаются два сигнала: инвертированный сигнал  $a_1$  и сигнал  $b_1$ . На входы конъюнктора  $F_2$  подаются сигналы  $a_1$  и  $b_1$ , но в этом случае сигнал  $a_1$  не инвертированный, а сигнал  $b_1$  инвертированный. Также по два сигнала подаются на конъюнкторы  $F_3$  и  $F_4$ .

Неиспользуемые сигналы на входы конъюнкторов  $F_1 \dots F_4$  не подаются. Для этого разрушаются оптические связи между входами устройства, выходами инверторов и входами конъюнкторов  $F_1 \dots F_4$ :

$$c_5 = D_5(F_1(\bar{a}_1 b_1), F_2(a_1 \bar{b}_1), F_3(\bar{a}_0 b_1), F_4(b_1 \bar{b}_0))$$

Оптический конъюнктор  $F_1$  выдает оптический сигнал в  $K$  условных единиц на выход, если у него на входе комбинация сигналов равна 01, а оптические конъюнкторы  $F_2, F_3, F_4$  выдают оптический сигнал в  $K$  условных единиц на свой выход, если у них на входе присутствует соответственно комбинация сигналов 10, 01 и 10. Пройдя оптический разветвитель  $S_i$ , оптический сигнал уменьшится и с интенсивностью 1 усл. ед. поступит на вход оптического дизъюнктора  $D_5$ . В данном случае один из конъюнкторов  $F_1 \dots F_3$  выдает сигнал высокого уровня интенсивности  $K$  усл. ед. на свой выход. Так как интенсивность сигнала на входе дизъюнктора  $D_5$  больше 0,5, то интенсивность сигнала на его выходе будет равна 1 усл. ед. В результате интенсивность сигнала на входе дизъюнктора  $D_5$  будет равна 1.

Для формирования сигнала  $c_4$  используется 4 конъюнктора  $F_5 \dots F_8$ . На входы конъюнкторов  $F_5$  и  $F_6$  входные сигналы подаются парами. На входы конъюнктора  $F_5$  подаются сигналы: инвертированный  $a_1$  и  $b_0$ , а на входы конъюнктора  $F_6$  подаются сигналы  $a_0$  и инвертированный сигнал  $b_0$ .

На входы конъюнкторов  $F_7$  и  $F_8$  подается по три входных сигнала. На входы конъюнктора  $F_7$  подается инвертированный сигнал  $a_0$  и сигналы  $b_1$  и  $b_0$ , а на входы конъюнктора  $F_8$  подаются сигналы  $a_1, a_0$  и инвертированный сигнал  $b_0$ :

$$c_4 = D_4(F_5(\bar{a}_1 b_0), F_6(a_0 \bar{b}_1), F_7(\bar{a}_0 b_1 b_0), F_8(a_1 a_0 \bar{b}_0))$$

Если на входе устройства присутствуют необходимые оптические сигналы, то оптический конъюнктор  $F_i$  выдает оптический сигнал в  $K$  условных единиц на свой выход.

Схема формирования сигналов дробной части  $c_3, c_2, c_1, c_0$  работает аналогично.

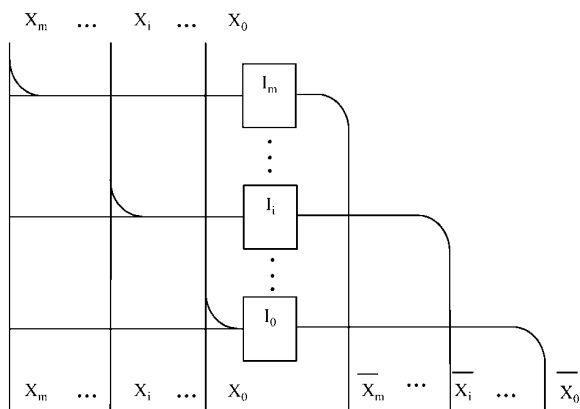


Рис. 2. Оптические инверторы

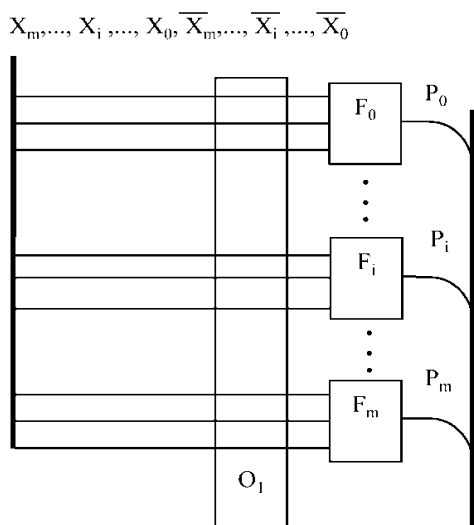


Рис. 3. Оптические конъюнкторы

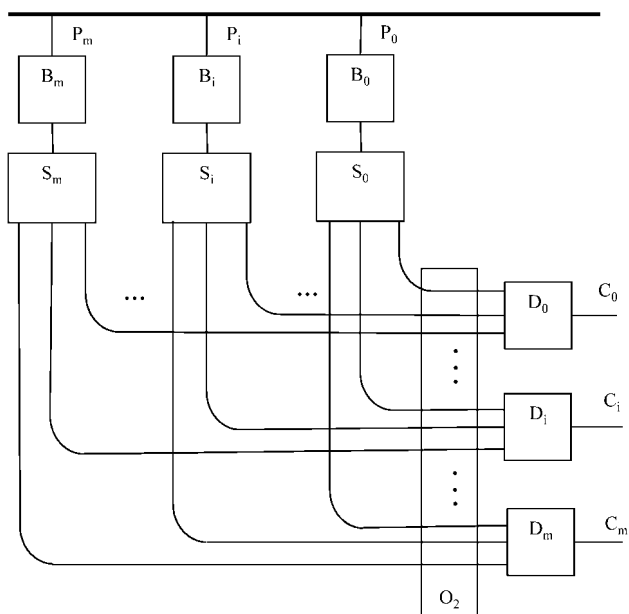


Рис. 4. Оптическая логическая матрица

### Программирование оптического вычислителя

В исходном состоянии все оптические связи со входами оптических конъюнкторов  $F_i$  не разрушены и на входы оптических конъюнкторов  $F_i$  подаются все входные сигналы  $X_0...X_m$  и все сигналы с выходов оптических инверторов  $I_i$ . При программировании оптического вычислителя для реализации конкретной функции, во-первых, производится разрушение оптических связей, обеспечивающих прохождение оптических сигналов на входы оптических конъюнкторов  $F_i$  для сигналов, не являющихся аргументами логических функций, а во-вторых, производится разрушение оптических связей, обеспечивающих прохождение оптических сигналов с выходов оптических разветвителей  $S_i$  на входы оптических дизъюнкторов  $D_i$ .

Например, для формирования сигнала  $c_6$  используется оптический конъюнктор  $F_0$ , на который до программирования подаются сигналы  $a_1, a_0, b_1, b_0, \bar{a}_1, \bar{a}_0, \bar{b}_1, \bar{b}_0$ . Сигналы  $\bar{a}_1, \bar{a}_0, \bar{b}_1, \bar{b}_0$  при формировании функции данного разряда не используются, поэтому оптические связи между выходами инверторов и входами оптического конъюнктора  $F_0$  разрушаются. Кроме этого, для формирования сигнала  $c_6$  используется оптический дизъюнктор  $D_0$ , на который до программирования подаются сигналы с выходов оптических разветвителей  $S_0-S_m$ . Оптические связи между выходами оптических разветвителей  $S_i-S_m$  и входами оптического дизъюнктора  $D_0$  не используются при формировании функции данного разряда, поэтому эти оптические связи также разрушаются.

Таким образом, метод разработки оптического программируемого вычислителя состоит из следующих этапов:

- исходя из параметров технической задачи, определяется сложная функция, для которой требуется производить вычисления;
- определяется диапазон входных аргументов;
- определяется требуемый диапазон выходных значений;
- определяется необходимая точность вычислений;
- определяется требуемое количество бит в целой и дробной части;
- производится вычисление значений функции;
- производится перевод значений из десятичной системы счисления в двоичную с фиксированной запятой;
- для каждого выходного бита определяется зависимость выходного значения от входных значений;
- производится оптимизация функций на основе законов алгебры логики, методами Карно и др.;
- осуществляется проверка полученных зависимостей;
- осуществляется построение оптической схемы путем удаления неиспользуемых оптических связей.

**Пример.** Для проверки правильности логических функций была написана программа на языке Си++. Программа осуществляет перебор всех входных комбинаций. Для каждой входной комбинации сигналов  $a_1, a_0, b_1, b_0$  осуществляется вычисление логических функций для каждого выхода устройства  $c_6, c_5, c_4, c_3, c_2, c_1, c_0$ .

```
for(i=0; i<=15; i++)
{
    a1 = (i&0x08) >> 3; a0 = (i&0x04) >> 2;
    b1 = (i&0x02) >> 1; b0 = (i&0x01);
}
```

```

c6 = a1&a0&b1&b0;
c5 = (~a1&b1) | (a1&~b1) | (~a0&b1) |
(b1&~b0);
c4 = (~a1&b0) | (a0&~b1) | (~a0&b1&b0) |
(a1&a0&~b0);
c3 = (a1&~a0&b1) | (a1&b1&~b0);
c2 = (~a1&a0&~b1&b0) | (~a1&a0&b1&~b0) |
(a1&~a0&~b1&b0) | (a1&~a0&b1&~b0)
| (a1&a0&b1&b0);
c1 = (~a1&a0&b0) | (a1&~a0&b1) |
(a0&~b1&b0) | (a1&b1&~b0);
c0 = (~a1&a0&b0) | (a0&~b1&b0);

std::cout << a1 << " " << a0 << " " <<
b1 << " " << b0 << " ";
std::cout << c5 << " " << c4 << " " <<
c3 << " " << c2 << " " << c1 << " "
<< c0 << " \n";
}

```

Реализация данной программы подтвердила эффективность предложенного подхода.

### Заключение

Проведен обзор существующих алгоритмов извлечения квадратного корня как базовой операции многих математических вычислений.

В ходе исследования изложены теоретические основы построения быстродействующего вычислителя квадратного корня для чисел, представленных в двоичном коде. Предложенный способ позволяет производить вычисление квадратного корня на основе комбинационного устройства, без использования метода проб и ошибок, что значительно увеличивает скорость вычислений при получении результата. Устройство может быть легко модифицировано для вычисления других функций или набора функций.

Разработана функциональная схема оптического программируемого устройства для выполнения функциональных преобразований. Рассмотрена работа составных элементов и всего устройства в целом.

Изложены принципы программирования оптического вычислителя для реализации конкретной функции.

Минимальное количество элементов и высокая скорость вычислений делают предложенное оптическое программируемое устройство перспективным при построении технических средств вычислительной техники и систем связи.

### Литература

1. Стаценко И.В. Исследование скорости сходимости одного обобщенного ньютоновского метода и классического метода Ньютона в процедуре уточнения корней многочлена // Точная наука. 2020. №78. С. 2-9.
2. Мачарадзе Г.Т., Морозова Е.А. Алгоритм быстрого вычисления квадратного корня и его актуальность // Вестник Российского нового университета. 2018. С. 106-109 DOI: 10.25586/RNU.V9187.18.04.P.106.

3. Ушенина И.В. Реализация на ПЛИС и сравнительный анализ вычислителей сигмоида, работающих с полным диапазоном аргумента с учетом симметрии // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2025. №71. С. 120-129. doi: 10.17223/19988605/71/1.

4. Гашков С.Б., Фролов А.Б., Попова Е.П. Об оценке сложности алгоритмов извлечения квадратных корней в конечных полях и кольцах вычетов // Вестник МЭИ. №5. 2018. С. 79-88.

5. Фёдоров А., Васнецова Т. Анализ эффективности методов извлечения корней // Вопросы математики, выпуск 1, 2023. С. 15-25.

6. Соколов Д. Новые подходы к извлечению корней // Труды Международной конференции по математике, 2024. С. 35-45.

7. Краснова А. Извлечение корней: эволюция методов // Электронный журнал «Математика XXI века», 2025. С. 12-20.

8. Златопольский Д.М. Методика извлечения квадратного и кубического корней в двоичной системе исчисления // Информатика в школе. 2021. № 1(1). С. 42-45 [https:// doi.org/10.32517/2221-1993-2021-20-1-42-45](https://doi.org/10.32517/2221-1993-2021-20-1-42-45).

9. Николаев С. Классические и современные методы извлечения корней. М.: Издательство: Мир науки, 2024, 320 с.

10. Сергеева О. Простота и сложность извлечения корней. М.: Издательство: Альфа-Бук, 2025, 200 с.

11. Andraos K., Ghiasi N.M. A High-Throughput FPGA Implementation of a Digit-Recurrence Square Root Unit with BFloat16 Support // IEEE Transactions on Computers. 2024. Vol. 73, No. 5, pp. 1245-1258. DOI: 10.1109/TC.2024.3355678

12. de Aledo M.R.G., de Guzmán P.L. A Formal Verification of the Fast Inverse Square Root Algorithm in the Coq Proof Assistant // Journal of Automated Reasoning. 2023. Vol. 67, Issue 4, Article 28. DOI: 10.1007/s10817-023-09670-6

13. Gupta S., Martinez A., Roberts M. Numerical Stability Analysis of Iterative Methods for Square Root in Mixed-Precision ML Training // Proceedings of the Machine Learning and Systems Conference (MLSys). 2024. Vol. 6, pp. 45-59.

14. Kaur J., Nguyen T.H., Bruguera J.D. Efficient Square Root and Inverse Square Root for the Posit Number System // IEEE Transactions on Emerging Topics in Computing. 2023. Vol. 11, No. 2, pp. 456-469 DOI: 10.1109/TETC.2023.3245671

15. Petrov D., Takahashi Y., Mosca M. A Scalable Quantum Algorithm for Square Root and Division with Arbitrary Precision // Quantum. 2024. Vol. 8, p. 1245. DOI: 10.22331/q-2024-01-08-1245

16. Wang C., Kim Y., Fahmy S.A. Hardware Optimizations of the Newton-Raphson Square Root for 3D Graphics Pipelines // ACM Transactions on Graphics (TOG). 2024. Vol. 43, No. 4, Article 78 DOI: 10.1145/3654662

17. Zhang F., Li H., McCluske E.J. Error Analysis and Correction for Digit-by-Digit Square Root in Safety-Critical Systems // IEEE Transactions on Computers. 2023. Vol. 72, No. 9, pp. 2541-2554. DOI: 10.1109/TC.2023.3267890

18. Pshko A. Mohammed, Integration of self-standing X- and Y-shaped polymer coupler and splitter with single mode optical fiber – Optical Materials <https://doi.org/10.1016/j.optmat.2020.110685>

19. Каменский В.В., Соколов С.В. Синтез преобразователей кодов на основе оптических технологий // Радиотехника. 2023. Т. 87. №11. С. 202-209.

20. Каменский В.В., Соколов С.В., Лагунова Е.О. Оптический цифро-аналоговый вычислитель // Т-Comm: Телекоммуникации и транспорт. 2025. Т. 19. № 4. С. 37-44.

21. Каменский В.В., Соколов С.В., Лагунова Е.О. Метод построения оптических вычислителей // Т-Comm: Телекоммуникации и транспорт. 2025. Т. 19. № 6. С. 33-42.

## FAST OPTICAL COMPUTERS

**Vladislav V. Kamenskij**, Rostov State Transport University (RSTU), Rostov-on-Don, Russia, [kam-vladislav@yandex.ru](mailto:kam-vladislav@yandex.ru)  
**Sergei V. Sokolov**, Moscow Technical University of Communications and Informatics, Moscow, Russia, [s.v.s.888@yandex.ru](mailto:s.v.s.888@yandex.ru)  
**Elena O. Lagunova**, Rostov State Transport University (RSTU), Rostov-on-Don, Russia, [lagunova@rambler.ru](mailto:lagunova@rambler.ru)

**Abstract**

The paper discusses a new approach to the design of high-performance computing systems based on optoelectronic technologies. Optical information processing devices are used in areas where maximum speed and simplicity of calculations are required, such as the synthesis of embedded systems and specialized processors for high-speed processing. In general, embedded systems process incoming data within a specific and predefined range, and calculations can be performed with a specified accuracy, but the calculation time is highly limited and critical. In this regard, for the developed optical calculator, as a result of conducting an analysis of existing formats of number representation, the format of number representation with a fixed point was chosen. As an example of the implemented nonlinear function, one of the most difficult functions for calculation in traditional schemes – the function of extraction of the square root was chosen. The theoretical foundations of building a fast-acting square root calculator for numbers represented in binary code are presented. The proposed approach allows for the calculation of the square root based on a combinational device, without the use of trial and error, which radically increases the speed of calculations. Since practical problems require a minimum amount of time to perform a set of calculations on several arguments, a functional diagram of a specialized optical programmable device has been developed to perform a set of functional transformations based on the principles of building and optimizing combinational circuits. The main basic elements of an optical programmable device are optical splitters, combiners, and transparencies, the technology of which is known and does not cause difficulties. Programming an optical computer to implement a specific function is done by destroying the optical. The minimum number of elements and high calculation speed make the optical programmable device promising for building fast-acting technical means of computer engineering and communication systems.

**Keywords:** optical computing device, square root extraction, exponentiation, gigahertz range, optical splitter, transparent, optical combiner.

**References**

- [1] I.V. Statsenko, "Research of the convergence rate of one generalized Newton method and the classical Newton method in the procedure of refining the roots of a polynomial," *Precise Science*. 2020. No. 78, pp. 2-9.
- [2] G.T. Macharadze, E.A. Morozova, "The algorithm of fast calculation of the square root and its relevance," *Bulletin of the Russian New University*. 2018, pp. 106-109 DOI: 10.25586/RNU.V9I87.18.04.P.106.
- [3] I.V. Ushenina, "Implementation on FPGA and comparative analysis of the sigmoid calculators working with the full range of the argument taking into account the symmetry," *Bulletin of Tomsk State University. Management, computer science and computer engineering*. 2025. No. 71, pp. 120-129. doi: 10.17223/19988605/71/1.
- [4] S.B. Gashkov, A.B. Frolov, E.P. Popova, "On the complexity of algorithms for extracting square roots in finite fields and residue rings," *Vestnik MEI*. No. 5. 2018, pp. 79-88.
- [5] A. Fedorov, T. Vasnetsova, "Analysis of the Effectiveness of Root Extraction Methods," *Journal "Issues of Mathematics"*, Issue 1, 2023, pp. 15-25.
- [6] D. Sokolov, "New Approaches to Root Extraction," *Proceedings of the International Conference on Mathematics*, 2024, pp. 35-45.
- [7] A. Krasnova, "Extraction of Roots: Evolution of Methods," *Electronic Journal "Mathematics of the 21st Century"*, 2025, pp. 12-20.
- [8] D.M. Zlatopolsky, "Methodology for Extracting Square and Cube Roots in the Binary Number System," *Informatics at School*. 2021. No. 1(1), pp. 42-45, <https://doi.org/10.32517/2221-1993-2021-20-1-42-45>.
- [9] C. Nikolaev, "Classical and Modern Methods of Root Extraction," Publisher: World of Science, 2024, 320 p.
- [10] O. Sergeeva, "Simplicity and Complexity of Root Extraction," Publisher: Alfa-Book, 2025, 200 p.
- [11] K. Andraos, N. M. Ghiasi, "A High-Throughput FPGA Implementation of a Digit-Recurrence Square Root Unit with BFloat16 Support," *IEEE Transactions on Computers*. 2024. Vol. 73, No. 5, pp. 1245-1258. DOI: 10.1109/TC.2024.3355678
- [12] M. R. G. de Aledo, P. L. de Guzman, "A Formal Verification of the Fast Inverse Square Root Algorithm in the Coq Proof Assistant," *Journal of Automated Reasoning*. 2023. Vol. 67, Issue 4, Article 28. DOI: 10.1007/s10817-023-09670-6
- [13] S. Gupta, A. Martinez, M. Roberts, "Numerical Stability Analysis of Iterative Methods for Square Root in Mixed-Precision ML Training," *Proceedings of the Machine Learning and Systems Conference (MLSys)*. 2024. Vol. 6, pp. 45-59.
- [14] J. Kaur, T. H. Nguyen, J. D. Bruguera, "Efficient Square Root and Inverse Square Root for the Posit Number System," *IEEE Transactions on Emerging Topics in Computing*. 2023. Vol. 11, No. 2, pp. 456-469 DOI: 10.1109/TETC.2023.3245671
- [15] D. Petrov, Y. Takahashi, M. Mosca, "A Scalable Quantum Algorithm for Square Root and Division with Arbitrary Precision," *Quantum*. 2024. Vol. 8, p. 1245. DOI: 10.22331/q-2024-01-08-1245
- [16] C. Wang, Y. Kim, S. A. Fahmy, "Hardware Optimizations of the Newton-Raphson Square Root for 3D Graphics Pipelines," *ACM Transactions on Graphics (TOG)*. 2024. Vol. 43, No. 4, Article 78 DOI: 10.1145/3654662
- [17] F. Zhang, H. Li, E. J. McCluske, "Error Analysis and Correction for Digit-by-Digit Square Root in Safety-Critical Systems," *IEEE Transactions on Computers*. 2023. Vol. 72, No. 9, pp. 2541-2554. DOI: 10.1109/TC.2023.3267890
- [18] Pshko A. Mohammed, Integration of self-standing X- and Y- shaped polymer coupler and splitter with single mode optical fiber - *Optical Materials* <https://doi.org/10.1016/j.optmat.2020.110685>
- [19] V.V. Kamensky, S.V. Sokolov, "Synthesis of code converters based on optical technologies," *Radio engineering*. 2023. Vol. 87. No. 11, pp. 202-209.
- [20] V.V. Kamensky, S.V. Sokolov, E.O. Lagunova, "Optical digital-analog computer," *T-Comm*. 2025. Vol. 19. No. 4, pp. 37-44.
- [21] V.V. Kamensky, S.V. Sokolov, E.O. Lagunova, "Method of constructing optical computers," *T-Comm*. 2025. Vol. 19. No. 6, pp. 33-42.