

Разработка и моделирование работы квантовой микроархитектуры с использованием алгоритма оптимизации квантового вычислительного процесса

В. С. Потапов¹, С. М. Гушанский

В настоящее время активно развивается теория квантовых вычислений. Несмотря на то, что квантовое вычислительное устройство обладает некоторыми особенностями, любую задачу, предназначенную для классической ЭВМ, можно воспроизвести и в квантовой системе. Целью исследования является разработка методики для построения моделей квантовых систем с помощью симуляторов. Предметом исследования являются методы построения квантовых вычислительных устройств и систем. В процессе исследования использовались методы системного анализа и компьютерного моделирования, методы объектно-ориентированного проектирования. В решении задач построения модульной квантовой системы с архитектурой открытого типа были использованы современные языки высокого уровня. Разработана и промоделирована работа квантовой микроархитектуры с использованием алгоритма оптимизации квантового вычислительного процесса. Выведена универсальная методология моделирования алгоритмов квантовой природы с использованием аппаратного ядра и требований к взаимной работе программной и аппаратной составляющих для эффективной работы квантовой системы.

Ключевые слова: квантовый алгоритм, квантовый бит, модель квантового вычислителя, квантовая схема, запутанность, квантовая схема, регистр, квантовое распознавание, гейт, параллелизм, интерференция, квантовый компьютер.

1. Введение

В настоящее время активно развивается теория квантовых вычислений [1]. Квантовое вычислительное устройство имеет ряд преимуществ перед классическими компьютерами, главным из которых является квантовый параллелизм вычислительного процесса. Данный квантовый эффект предлагает одновременное вычисление с набором разных исходных данных. Ряд квантовых алгоритмов способен вычислять некоторые задачи на порядок эффективнее своих классических аналогов. В настоящее время существует несколько модельных устройств, использующих кубиты для вычислений. Актуальность темы квантовых компьютеров подтверждается тем, что уже сейчас банковские системы, криптография и шифрование информации требуют колоссальных ресурсов вычислительных систем. Это только некоторые примеры, но наиболее важные и яркие, которые касаются повседневной жизни огромной части населения планеты. Процесс упирается в физику, велика вероятность, что меньших техпроцессов в привычных системах уже не увидать. В этот момент требуется менять логику работы вычислительной системы, уходить от обыденной двоичной в новую сферу, в вычисления, основанные на квантовой механике.

¹ Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 20-07-00916.

2. Теория квантовых вычислений

В квантовых вычислениях существует понятие квантового бита (кубита) [2], содержащего в себе суперпозицию (взаимосвязь) нуля и единицы [3]. Пусть имеем вычислительную систему [4], состоящую из двух базисных состояний $|0\rangle$ и $|1\rangle$. Один кубит может находиться в любой суперпозиции $\alpha_0|0\rangle + \alpha_1|1\rangle$, $|\alpha_0|^2 + |\alpha_1|^2 = 1$. Соответственно, 1 кубит находится в векторном пространстве C^2 . Аналогичным образом можем думать о системах из более чем одного кубита, которые существуют в пространстве тензорных произведений нескольких систем кубитов. В общем, регистр из n кубитов имеет 2^n базисных состояний [5], каждое из которых имеет вид $|b_1\rangle \otimes |b_2\rangle \otimes \dots \otimes |b_n\rangle$, где $b_i = \{0,1\}$. Можем сократить это до $|b_1b_2\dots b_n\rangle$. Отсюда следует, что базисные состояния двух кубитов $|x\rangle$ и $|y\rangle$ ортогональны, если $x \neq y$. Другой способ увидеть эту ортогональность – использовать правила тензорного произведения $\langle x|y\rangle = \langle x_1|y_1\rangle \otimes \dots \otimes \langle x_n|y_n\rangle = \langle x_1|y_1\rangle \dots \langle x_n|y_n\rangle$. Один кубит, описываемый с помощью волновой функции, в матричной форме может быть представлен как вектор-столбец вида $[\alpha \ \beta]^T$. Здесь стоит заметить, что для вычислений необходимо знать только амплитуды состояний [6]. Вектор-столбец состояний представляет собой при моделировании квантовый регистр [7]. В общем случае квантовый регистр описывается как:

$$|\psi\rangle = |\psi_1, \psi_2, \dots, \psi_n\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \otimes \dots \otimes |\psi_n\rangle, \quad (1)$$

где $\psi_i, i = 1 \dots n$ – волновые функции [8] отдельно взятых кубитов, а n – общее количество кубитов в моделируемой квантовой системе.

3. Представление квантовых вентилях при помощи квантовых схем

Квантовые схемы [9] – это еще один инструмент для описания операций над квантовым регистром кроме формального способа описания процесса квантовых вычислений. Они наиболее наглядно позволяют проследить изменения квантового регистра под воздействием квантовых вентилях.

Квантовые схемы представляют собой набор линий (кубитов) и вентилях (прямоугольника, у которого изображается первая заглавная буква названия вентиля), которые отображают эволюцию (изменение) квантовой системы. Кубиты заранее находятся в определенных состояниях на входах схемы. Затем происходит эволюция системы кубитов посредством квантовых вентилях. Эволюция [10] заканчивается, когда систему необходимо измерить. Пример такой схемы, которая производит запутывание двух кубитов, изображен на рис. 1.

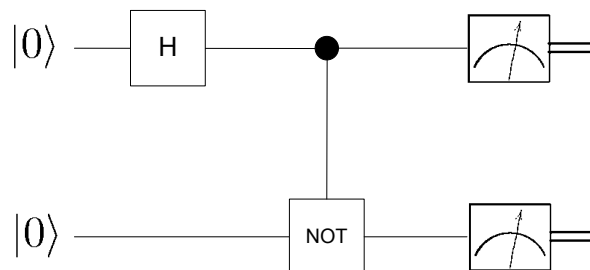


Рис. 1. Пример квантовой схемы запутывания двух кубитов

Далее для описания квантовых вентилях наряду с формальным представлением также будут использоваться соответствующие квантовые схемы вентилях. Способы манипуляции квантовым регистром представляют по аналогии с классической архитектурой квантовые вентилях. Они отличаются от обычных логических вентилях (гейтов) тем, что работают с ку-

битами, а следовательно, подчиняются квантовой логике. Квантовые вентили в отличие от многих классических всегда являются обратимыми.

4. Алгоритм оптимизации, исключая тензорное произведение при моделировании

Следующий алгоритм применяется в программных ядрах различных типов, а также в системах, состоящих из нескольких процессоров, для уменьшения количества итераций работы. Возьмем уравнение и покажем на его примере суть данного оптимизационного алгоритма. Матрица состоит из огромного числа элементов-нулей, не несущих информационной нагрузки. Основной задачей предложенного метода является вывод матрицы 2x2 в рамках однокубитового вентилей [11] перемножения ее и вектора квантового состояния с помощью определенной последовательности шагов. Пусть имеем 3 квантовых бита, тогда описанная выше последовательность будет соответствовать приведенной на рис. 2. За отсутствием оптимизации необходимо произвести единственное взаимодействие с регистрами за время $O(2^{4n} + 2^{3n})$ и использовать $O(2^{2n} + 2^n)$ ячеек, n – количество кубитов. При использовании оптимизации набор операций будет соизмеримым с $O(2^{n-1})$, а объем памяти будет составлять $O(2^n)$. Для программного представления и работы с разреженными матрицами используется модуль Sparse из библиотеки SciPy (Python), предназначенной для выполнения научных и инженерных расчётов. Для непосредственной работы с матрицами используется Йельский формат, предполагающий сжатое хранение строкой и использование трех массивов данных (массив значений, массив индексов столбцов, массив индексации строк). На рис. 2 показана процедура выборки пары состояний из модели, в которую попадают ненулевые элементы матриц в соответствии с номером квантового бита.

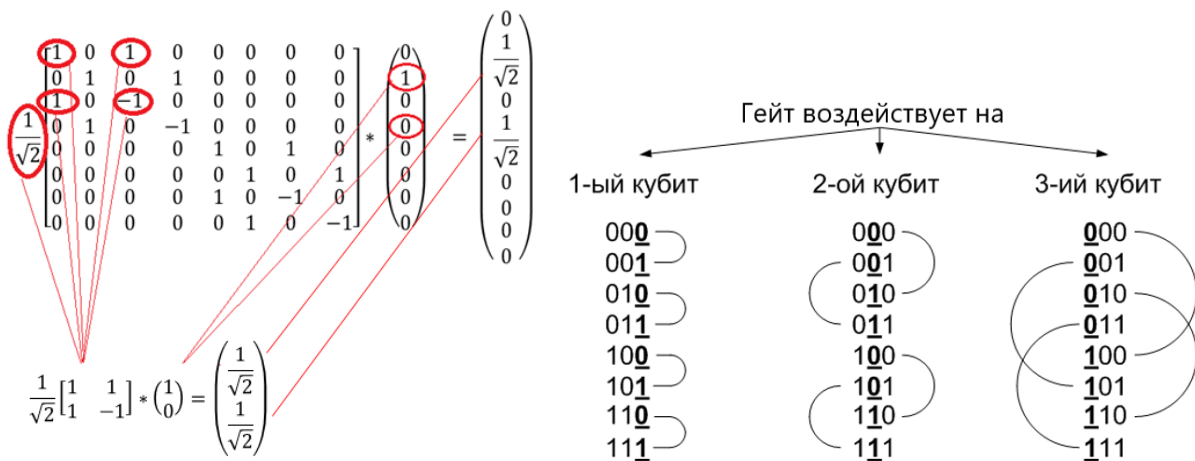


Рис. 2. Алгоритм оптимизации; порядок выбора квантового состояния по номеру квантового бита

Представим алгоритм, позволяющий увеличивать производительность квантового вычислительного устройства для операций с квантовыми битами (рис. 3). В первую очередь выполняется инициализация квантового регистра, то есть происходит установка амплитуды ($Amr = 1$) для любого состояния регистра. Далее вычисляется бинарный индекс квантового бита, представляющего числовое значение с установлением единицы в разряде, который соответствует порядковому номеру квантового бита, с которым производится взаимодействие. После циклично выполняется перебор состояний квантового регистра, а также вывод пары состояний, осуществляющей работу с вентилем. Для верного вывода применяется переменная маска, фильтрующая набор индексов состояния для соблюдения последовательности получения пар состояний. Алгоритм оканчивается после перебора всех квантовых состояний.

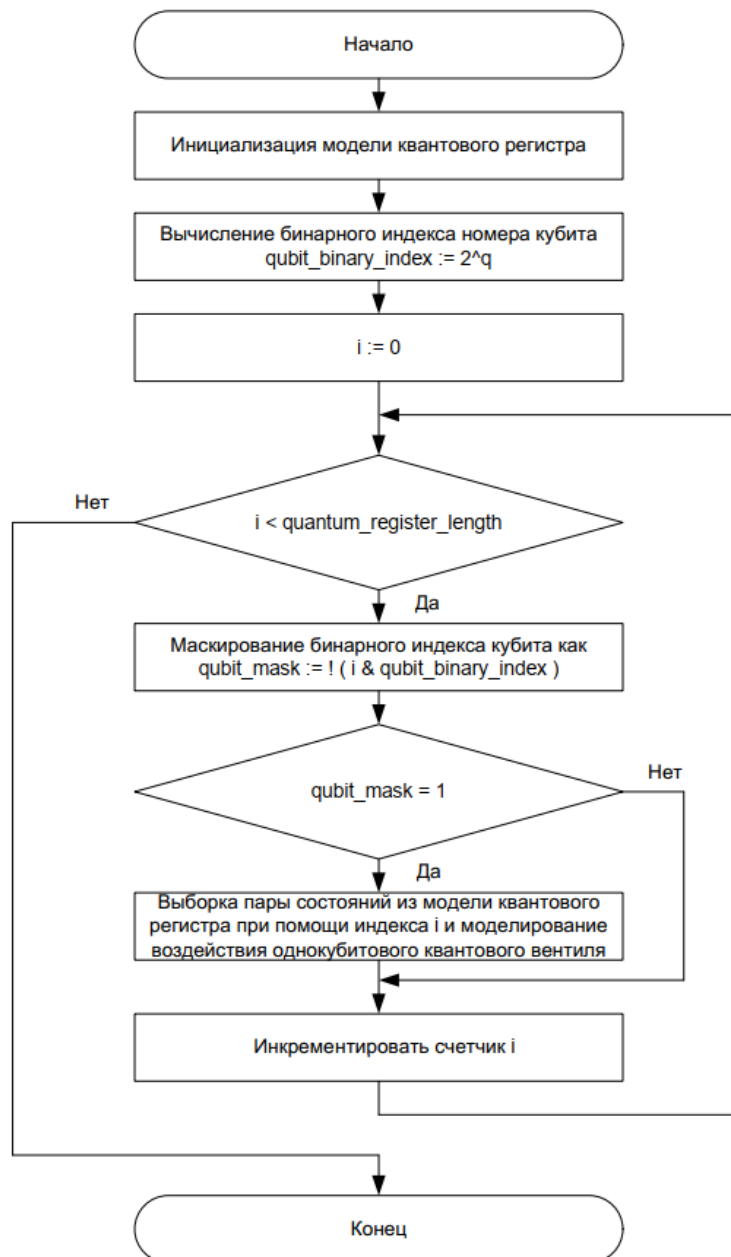


Рис. 3. Блок-схема оптимизационного алгоритма взаимодействия с однокубитовыми вентилями

5. Сравнительный анализ оптимизационного алгоритма моделирования квантовых вычислений и математического моделирования

В работе предложен оптимизационный алгоритм, позволяющий исключать произведение тензорной природы во время процесса моделирования. Алгоритм применяется в ядрах программ, приложений и в многопроцессорных информационных системах с целью уменьшения количества операций, которые взаимодействуют с тензорным произведением, абсолютным удалением преобразовательных матриц. Проверим эффективность оптимизационного алгоритма при помощи программ сравнения результатов процесса моделирования, далее сравниваем реализацию алгоритма с помощью программы или ускорителя. При математическом моделировании [12] квантового вычислительного процесса требуется совершить операцию умножения тензорного и матричного типов. Оптимизационный алгоритм наиболее предпочтителен с точки зрения сложности вычислений и объема памяти. Таблица 1 показывает ре-

зультаты процесса моделирования одного воздействия квантового вентиля на квантовые регистры с разным количеством квантовых битов.

Таблица 1. Результаты моделирования однокубитового вентиля на квантовый регистр с помощью программных средств

Число кубитов	Время выполнения однокубитового воздействия по методике математического моделирования, нс	Время выполнения однокубитового воздействия с оптимизацией, нс
3	21025	7446
4	46432	10512
5	137545	18397
6	539668	30662
7	2049165	54754
8	8273320	102941
9	31202690	204127
10	53175812	408694

Как видно из табл. 1, использование алгоритма повышает эффективность процесса моделирования для однокубитовых вентилях, что делает его использование оправданным и необходимым при построении аппаратного симулятора.

6. Сравнение программной реализации алгоритма оптимизации с аппаратной реализацией

Для сравнения программной реализации алгоритма оптимизации с его аппаратной версией для однокубитовых квантовых вентилях, имея модель САПР Quartus II, можно привести следующие расчеты. Количество тактов блока генерации пар состояний для однокубитовых вентилях составляет $q + 2^{n-1} * 5$, где q тратятся на сдвиги при инициализации и 2^{n-1} тактов на генерацию пар состояний, q – номер кубита, n – количество кубитов в квантовом регистре, 5 тактов тратится на вычисление следующего индекса состояния.

Данное количество тактов получается вследствие перемножения матрицы однокубитового квантового вентиля на пару значений из вектора состояний модели квантового регистра. Причем перемножаются комплексные числа. При чтении из ОЗУ тратится 3 такта, а при записи – 2. Используя данные значения, представим таблицу сравнения программной и аппаратной реализации моделей. Для аппаратуры будем считать, что частота ПЛИС равняется 130 МГц. Такая частота была оптимальной при моделировании в Quartus II на ПЛИС Stratix III EP3S3E50F780C4L модели квантового регистра из 5 кубитов.

Таблица 2. Результаты сравнения программного и аппаратного моделирования в случае однокубитовых вентилях

Число кубитов	Время выполнения однокубитового воздействия с оптимизацией, нс	Время выполнения однокубитового воздействия в аппаратной модели, нс
3	7447	4466
4	10513	8932
5	18398	17864
6	30663	35728
7	54755	71456
8	102940	140912
9	204128	285824
10	408695	455532

Получение выигрыша аппаратной реализации достигается за счет реализации матричного умножителя. При числе квантовых битов 5 или более аппаратное моделирование [13] уступает программному. Также стоит заметить, что частота процессора ПК заметно выше по сравнению с частотой ПЛИС. Поэтому можно дать рекомендации по использованию более мощной ПЛИС, поддерживающей частоту на уровне 200 МГц и выше. Без оптимизации мы вынуждены производить одно воздействие на квантовый регистр за время $O(2^{4n} + 2^{3n})$ и использовать $O(2^{2n} + 2^n)$ ячеек памяти, n – количество кубитов. Данный алгоритм хотя и входит в класс NP-сложных алгоритмов, однако его использование предпочтительно.

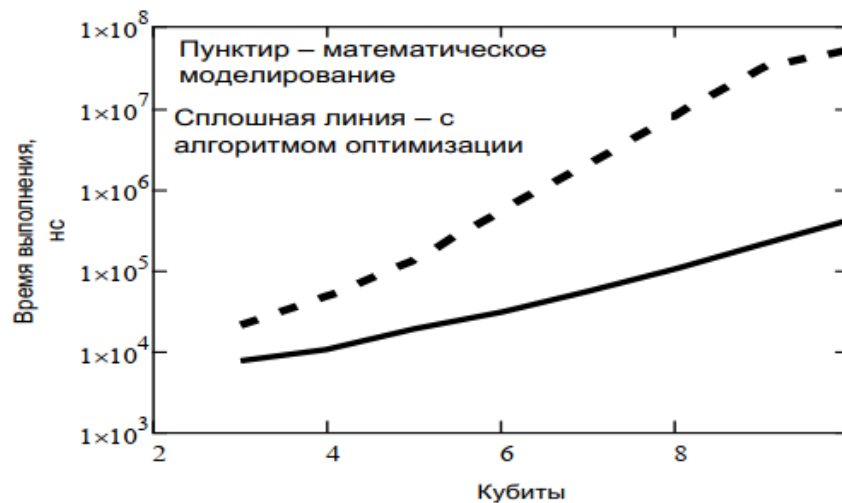


Рис. 4. Результаты моделирования

7. Заключение

В результате исследования подходов и методик к моделированию квантовых вычислений были выделены их преимущества и недостатки, а также предложена оригинальная методика моделирования квантового вычислительного процесса с помощью аппаратного ускорителя. Методика оправдала себя в экспериментальной части, для создания которой были заимствованы алгоритмы программных симуляторов.

Однако программные симуляторы пока не способны производить вычисления с упрощенными типами данных. По этой причине в рамках анализа математических компонентов квантового вычислительного процесса был определен тип хранения амплитуд квантовых состояний. Это позволяет сократить часть расходов на ресурсы хранения в моделях квантовых вычислительных систем [14]. Аппаратная составляющая позволяет создавать специальные операции, превышающие классические типы процессоров с точки зрения производительности. Если рассматривать ПЛИС в качестве программно-аппаратной основы для будущего квантового ускорителя, то становится возможным выделение гибкости и универсальности методики, не нацеленной на моделирование конкретного количества квантовых битов.

Выражение благодарности

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 20-07-00916.

Литература

1. *Feynman R. P.* Simulating physics with computers // International Journal of Theoretical Physics. 1982. V. 21, № 6. P. 467–488.
2. *Richter M., Arnold G., Trieu B., Lippert T.* Massively Parallel Quantum Computer Simulations: Towards Realistic Systems. John von Neumann Institute for Computing, NIC series. 2007. V. 38. P. 61–68.
3. *Khalid A. U.* FPGA Emulation of Quantum Circuits: master of Computer Engineering thesis: 31.10.2005. McGill University, 2005. 73 p.
4. *Alcazar J., Leyton-Ortega V., Perdomo-Ortiz A.* Classical versus Quantum Models in Machine Learning: Insights from a Finance Application // Phys. Re. Lett. 2020, arXiv: 1908.10778.
5. *Barends R. et al.* Coherent Josephson Qubit Suitable for Scalable Quantum Integrated Circuits // Phys. Re. Lett. 2013. arXiv:1304:2322. DOI: 10.1103/PhysRevLett.111. 080502.
6. *Shor P. W.* Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer // Foundations of Computer Science: Conference Publications. 1997. P. 1484–1509.
7. *Guzik V., Gushanskiy S., Polenov M., Potapov V.* Architecture and Software Implementation of a Quantum Computer Model // 5th Computer Science On-line Conference 2016 (CSOS), Czech Republic, 2016. P. 59–68.
8. *Guzik V.* Development of Methodology for Entangled Quantum Calculations Modeling in the Area of Quantum Algorithms // 6th Computer Science On-line Conference 2017 (CSOS), Czech Republic, 2017. P. 106–115.
9. *Barenco A., Bennett C. H., Cleve R. et al.* Elementary gates for quantum computation // Phys. Rev. A. 1995. V. 52, № 5. P. 3457–3467.
10. *Potapov V., Gushanskiy S., Polenov M.* The Methodology of Implementation and Simulation of Quantum Algorithms and Processes // IEEE 11th International Conference on Application of Information and Communication Technologies (AICT), 2017. P. 437–441.
11. *Gao C., Jiang D., Guo Y., and Chen L.* Multi-matrix error estimation and reconciliation for quantum key distribution // Opt. Express. 2019. № 27 (10), 14545.
12. *Biamonte J., Wittek P., Pancotti N., Rebentrost P., Wiebe N., and Lloyd S.* Quantum machine learning // Nature. 2017. № 549. P. 195–202.
13. *Tang E.* A quantum-inspired classical algorithm for recommendation systems // Proc. of the 51st Annual ACM SIGACT Symposium on Theory of Computing, 2019. P. 217–228.
14. *Abhijith J., and Patel A.* Spatial Search on Graphs with Multiple Targets using Flip-flop Quantum Walk // Quantum Information and Computation. 2018. № 18. P. 1295–1331.

*Статья поступила в редакцию 25.09.2021;
переработанный вариант – 06.11.2021.*

Потапов Виктор Сергеевич

ассистент кафедры вычислительной техники ИКТИБ ЮФУ (347900, Таганрог, ул. Энгельса, 1), тел. (8634) 37-17-56, e-mail: vpotapov@sfnedu.ru.

Гушанский Сергей Михайлович

к.т.н., доцент кафедры вычислительной техники ИКТИБ ЮФУ (347900, Таганрог, ул. Энгельса, 1), тел. (8634) 37-17-56, e-mail: smgushansky@sfnedu.ru.

Development and simulation of the quantum microarchitecture operation using an optimization algorithm for a quantum calculation process**V. Potapov, S. Gushansky**

The theory of quantum computing is being actively developed. Despite the fact that quantum computing device has some peculiarities, any task designed for a classical computer can be reproduced on a quantum system. The aim of the research is to develop a methodology for constructing models of quantum systems simulators using hardware simulators. The subject of this research is methods for constructing quantum computing devices and systems. In the course of the research, the system analysis methods and computer modeling, methods of object-oriented design were used. Modern high-level languages were used to solve the problems of constructing a modular quantum system with an open architecture. The work of a quantum microarchitecture using an algorithm for optimizing a quantum computational process has been developed and modeled. A universal methodology for modeling algorithms of a quantum nature using the hardware core and requirements for the mutual operation of software and hardware components for the efficient operation of a quantum system is derived.

Keywords: quantum algorithm, quantum bit, quantum computing model, quantum circuit, entanglement, quantum circuit, register, quantum recognition, gate, parallelism, interference, quantum computer.