УДК 004.315.2

# КВАНТОВЫЙ СУММАТОР ЧИСЕЛ С ПЛАВАЮЩЕЙ ТОЧКОЙ

**Краснобровкин П. С.**
**научный руководитель канд. техн. наук Кузнецов А. С.**
***Сибирский федеральный университет***

## Введение

В настоящее время популярность набирает класс обратимых вычислений, в особенности квантовые. Такие вычисления основаны на постулатах квантовой механики. Основной единицей хранения информации является кубит – элемент, способный находиться в суперпозиции двух состояний классического бита. Хотя задача построения компьютера, отвечающего всем требованиям квантовой механики, всё ещё ожидает своего решения, активно предпринимаются попытки создать компьютеры, по свойствам приближённые к квантовым. В качестве примера можно привести D-Wave Two [2].

Ввиду этого, уже можно составлять алгоритмы для квантовых компьютеров. Более того, уже созданы эмуляторы квантовых систем, а также высокоуровневый язык программирования [5], включающий в себя арифметику целых чисел, чисел с фиксированной запятой и несколько алгоритмов, основанных на этом.

Следующим этапом развития представляется разработка арифметики вещественных чисел, что позволит перейти к реализации алгоритмов достаточно широкого класса. Далее мы рассмотрим вариант алгоритма квантового сумматора двух вещественных чисел, представленных в формате IEEE-754. Выбор этого формата обусловлен тем фактом, что на сегодняшний день все микропроцессоры представляют вещественные числа, а также выполняют над ними операции в соответствии с этим стандартом.

## Постановка задачи

Рассматриваем два числа (A и B) с плавающей точкой, имеющие представление в 32-битном формате IEEE-754 [3]. Для разрабатываемого сумматора необходимы следующие допущения:

1. Порядок следования кубит – сверху вниз, от самого старшего к самому младшему;
2. Оба числа положительны;
3. На представление каждого из чисел потребуется 31 кубит, поскольку отсутствует необходимость в знаковом разряде;
4. Число A – наибольшее по порядку, его представление подаётся на вход первых 31 кубит;
5. Ведущие единицы нормализованных мантисс пишутся в явном виде в соответствующие старшие разряды;
6. Во время работы алгоритма не отслеживаются исключительные ситуации – “бесконечность” и “не число” (NaN). Соответствующие представления можно обнаружить только на момент снятия результата

Требуется выполнить суммирование этих чисел, результат поместить в кубиты числа A.

## Алгоритм работы

Сначала опишем в общем виде шаги работы квантового сумматора. Пояснения, а также иллюстрации ключевых моментов приведём в последующих разделах. Итак, основные шаги алгоритма следующие:

1. Применяются два квантовых преобразования Фурье к кубитам, отвечающим за порядок числа B и мантиссы A;
2. Из порядка B вычитается порядок A, знак результата не учитывается. К порядку B применяется обратное преобразование Фурье. В результате получаем количество сдвигов мантиссы B;
3. Поскольку ведущие единицы мантисс при суммировании дадут перенос, к порядку числа A сразу же прибавляется единица;
4. Мантисса числа B сдвигается вправо на число, записанное в кубитах порядка B;
5. Мантиссы A и B суммируются как целые числа;
6. К мантиссе числа A применяется обратное преобразование Фурье;
7. Мантисса числа A сдвигается на один разряд вправо. В старшем разряде в явном виде устанавливается значение “1”;
8. Кубиты, отвечающие за представление числа A, содержат результат, доступный для интерпретации в классическом виде.

Разберём ключевые моменты более подробно.

## Сумматор целых чисел

Будем рассматривать целочисленный сумматор, полученный в работе [1]. Суммирование целых чисел базируется на кодировании первого аргумента с помощью квантового преобразования Фурье. Введём следующие обозначения: H – вентиль Адамара, $R_k$ – вентиль условного вращения, $\phi_k(c)$ – состояние k-ого кубита входного числа a. Они вычисляются по следующим формулам:

$$H=\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix},\ R_k=\begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & \exp(2i\pi(\frac{1}{2^k}))\end{pmatrix},\ \phi_k(c)=\frac{1}{\sqrt{2}}\left(|0\rangle+\exp\left(2i\pi\frac{c}{2^k}\right)|1\rangle\right).$$

Схема квантового преобразования Фурье над некоторым целым числом “c” имеет следующий вид:

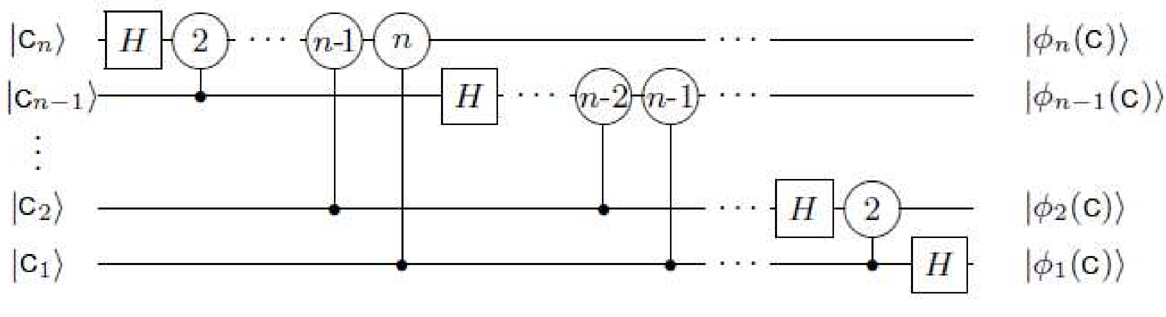


Рисунок 1 – Схема квантового преобразования Фурье.

Итоговое состояние квантовой системы (регистра) получается путём тензорных произведений состояний каждого кубита, начиная со старшего разряда.

Схема квантового целочисленного сумматора может иметь следующий вид:

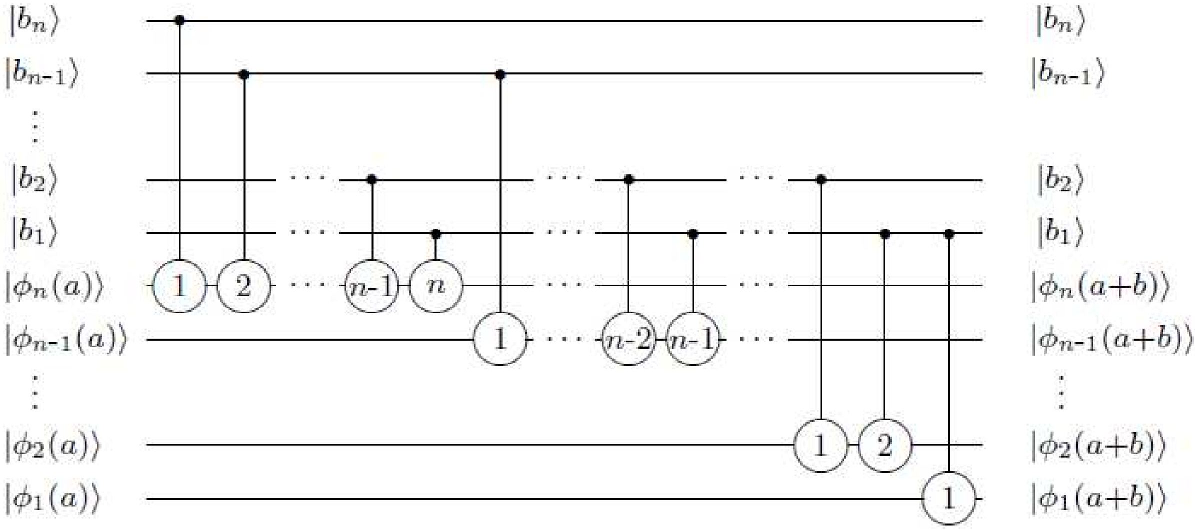


Рисунок 2 – Квантовый сумматор целых чисел.

Для представления результата в классическом виде требуется применить обратное преобразование Фурье. Поскольку квантовые вычисления относятся к классу обратимых вычислений, то эта задача сводится к выполнению схемы, представленной на рисунке 1, в обратном порядке, начиная с младшего кубита.

**Битовый сдвиг вправо**

Эта задача решается путём последовательного применения двухкубитного вентиля обмена (“Swap gate”), имеющего следующее матричное представление [4]:

$$SWAP = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Схема битового сдвига вправо аналогична циклическому сдвигу числа и имеет следующий вид:

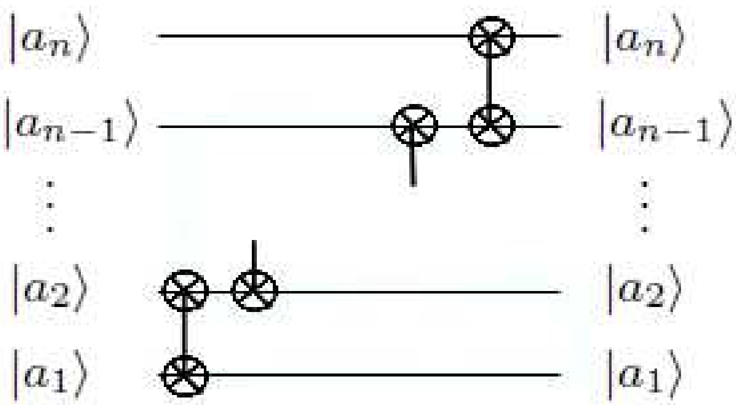


Рисунок 3 – битовый сдвиг вправо.

## Заключение

Предложен алгоритм работы квантового сумматора на примере чисел, имеющих представления в 32 битах. К его достоинствам можно отнести:

1. параллелизм на уровне некоторых основных этапов работы алгоритма в целом;
2. на большинстве участков схемы возможны параллельные изменения состояний нескольких кубит;
3. отсутствие затрат на вспомогательные кубиты;
4. возможность применения алгоритма к вещественным числам с представлениями произвольной длины. Для этого лишь требуется определить количество бит, отводимых на запись порядка числа и его мантиссы

Если решаемая задача накладывает требование на дальнейшее использование значений входных аргументов сумматора, то под результирующую сумму следует выделить регистр, называемый “аккумулятором”. Тогда алгоритм суммирования выглядит следующим образом:

1. Перед началом работы алгоритма суммирования для аккумулятора выделяются кубиты и в явном виде устанавливаются в нуль;
2. Происходит параллельное применение вентиля “исключающее или”[4] к k-ым кубитам аккумулятора и входного числа A;
3. Дальнейшие шаги совпадают с вышеописанным алгоритмом за тем исключением, что все необходимые вентили применяются к кубитам аккумулятора и входного числа B – происходит суммирование аккумулятора и B

Представляются следующие направления дальнейших научных исследований:

1. Разработка квантовых алгоритмов для операций вычитания, умножения, извлечения квадратного корня и тригонометрии над числами с плавающей точкой;
2. Обобщённые реализации вышеуказанных алгоритмов в рамках модуля языка Quipper. Такие реализации позволят рассматривать числа с представлениями произвольной длины в формате IEEE-754;
3. Портирование алгоритмов, предназначенных для классического компьютера, на квантовый;
4. Разработка архитектурно-независимого языка промежуточного представления квантовых программ;
5. Разработка трансляторов из уже существующих языков программирования (например, язык C) в язык промежуточного представления

## Список использованных источников


1. Addition on a Quantum Computer, Thomas G. Draper [Электронный ресурс] / 3 апрель 2014. – URL: http://arxiv.org/abs/quant-ph/0008033
2. D-Wave Two™ System. The first commercial quantum computer [Электронный ресурс] / 3 апрель 2014. – URL: http://dwavesys.com/d-wave-two-system
3. IEEE 754: Standard for Binary Floating-Point Arithmetic [Электронный ресурс] / 3 апрель 2014. – URL: http://grouper.ieee.org/groups/754/
4. Quantum Gate. Wikipedia – the free encyclopedia [Электронный ресурс] / 3 апрель 2014. – URL: http://en.wikipedia.org/wiki/Quantum_gate
5. Quipper Language [Электронный ресурс] / 3 апрель 2014. – URL: http://mathstat.dal.ca/~selinger/quipper