УДК 004.382

К-КУБИТНЫЙ ЦИФРОВОЙ КВАНТОВЫЙ СОПРОЦЕССОР НА ПЛИС

В. С. Глухов
Национальный университет «Львовская политехника»

Аннотация. *Приведены модели способного выполнять квантовое преобразование Фурье цифрового квантового сопроцессора, квантовых вентилей и кубитов. Работа сопроцессора была промоделирована, модели были имплементированы в ПЛИС, были определены их временные и аппаратные характеристики. Время выполнения одного квантового преобразования Фурье не зависит от количества цифровых кубит в сопроцессоре и соизмеримо с временем вычислений в аналоговом квантовом компьютере, аппаратная сложность линейно зависит от количества цифровых кубит.*

Ключевые слова: *цифровой квантовый сопроцессор, цифровой кубит, квантовое преобразование Фурье.*

Вступление

Классические квантовые компьютеры (КК) являются аналоговыми вычислительными устройствами [1]. Это вычислительные устройства, которые используют явления квантовой механики (квантовая суперпозиция, квантовая запутанность) для передачи и обработки данных. Квантовый компьютер (в отличие от обычного классического компьютера) работает не с битами (способными принимать 0 или 1), а с кубитами, каждый из которых в любой момент времени может иметь какое-то значение x (можно считать для простоты, что $0 \le x \le 1$), которое с точно определенной вероятностью p может измеряться как 0, и с точно определенной вероятностью $1\text{-}p$ – как 1 (квантовая суперпозиция). В ходе вычисления каждый кубит может принимать любое значение в обозначенном диапазоне, но в процессе измерения конечного результата эта суперпозиция с указанной вероятностью обращается в одно из двух классических состояний, 0 или 1.

Цифровой квантовый компьютер, а именно, классический процессор и его цифровой квантовый сопроцессор, который может быть реализован на ПЛИС, описан в [2]. Цифровой квантовый сопроцессор работает под управлением классического компьютера, и вместе они образуют цифровой квантовый компьютер. Цифровой квантовый сопроцессор представляет собой набор цифровых блоков, называемых цифровыми кубитами, каждый из которых имеет многобитный вход и однобитный выход. В цифровом кубите имеется генератор псевдослучайных кодов (*PRNG*) для генерации выходного бита, который может принимать значение 0 или 1 с вычисленной вероятностью.

Для некоторых задач число кубитов в квантовом компьютере является полиномиальной функцией сложности задачи. Например, известны алгоритмы факторизации n-битного целого числа с использованием чуть более $2n$ кубитов (алгоритм Шора [3]). Предполагается, что для взлома шифра *RSA* с 2048-битным ключом необходимо не менее 4000 кубитов (но может быть больше в зависимости от используемого алгоритма [4]).

Области применения квантовых компьютеров постоянно расширяются, известны предложения по их применению к обработке изображений [5]. Сама возможность их появления [4] заставляет искать новые решения казалось бы уже решенных задач [6].

Настоящие квантовые компьютеры являются аналоговыми и вероятностными устройствами. Их создание - очень сложная и дорогостоящая задача. Все это подчеркивает актуальность работ по изучению характеристик квантовых компьютеров, созданию их цифровых моделей и цифровых версий, а также подготовке специалистов для работы как с аналоговыми, так и с цифровыми квантовыми компьютерами.

В этой статье многокубитный цифровой квантовый сопроцессор проверяется на возможность выполнения квантового преобразования Фурье, которое является основной квантовой операцией алгоритма Шора.

Для этого были созданы модели цифровых квантовых сопроцессоров с числом кубит до 256, их работа была промоделирована и каждая из версий сопроцессора была реализована на одной ПЛИС.

1. Физические основы квантовых компьютеров

Если классический компьютер в любой момент может находиться только в одном из своих возможных состояний $|0\rangle, |1\rangle, ..., |N-1\rangle$ (в записи Дирака), то квантовый компьютер в каждый момент времени находится одновременно во всех этих основных состояниях, но в каждом состоянии $|j\rangle$ - со своей комплексной амплитудой λ_j . Это квантовое состояние называется «квантовой суперпозицией» классических состояний и называется волновой функцией $|\psi\rangle = \sum_{j=0}^{N-1} \lambda_i |j\rangle$. При этом вероятность определения состояния квантового компьютера как $|j\rangle$ в результате измерения состояний его кубитов равна $p_j = \lambda_j^2$. Сумма всех вероятностей равна $P = \sum_{j=0}^{N-1} \lambda_i^2 = 1$ [7].

Квантовое состояние может изменяться во времени двумя принципиально разными способами: с помощью унитарной квантовой операции (унитарного преобразования) и с помощью измерения [7]. Любое унитарное преобразование волновой функции можно представить или в виде простого смещения точки на поверхности сферы единичного радиуса (сфера Блоха для комплексных амплитуд, рис. 1, [5]),

Хорошей иллюстрацией поведения кубита может быть, спин электрона (рис. 2 [8]), который всегда направлен перпендикулярно направлению движения электрона. Из-за постоянного изменения орбиты движения электрона (рис. 3 [9]) его спин может быть произвольно направлен, что соответствует движению вектора в сфере Блоха.

Классические компьютеры – преимущественно электронные устройства. Носителем информации в них является заряд электрона. Теоретический предел такого способа представления информации является заряд одного электрона: есть электрон – логическая 1, нет электрона (дырка) – логический 0. Если использовать для представления информации телесный угол наклона оси вращения того же одного электрона, то тогда один электрон может нести не 1 бит информации, а больше (на сколько больше – зависит от точности определения угла наклона оси и точности управления им).

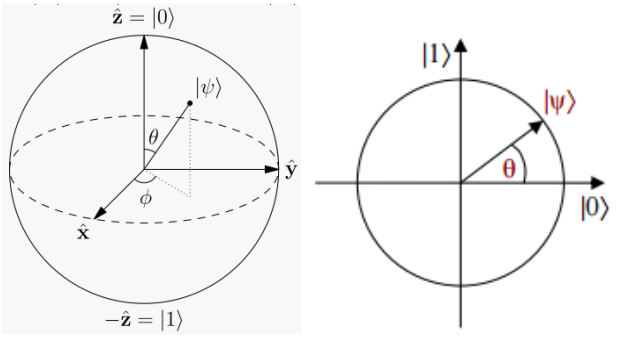


Рис. 1. Представление кубита в виде сферы Блоха для комплексных амплитуд (вверху) и единичного круга для вещественных амплитуд (внизу)

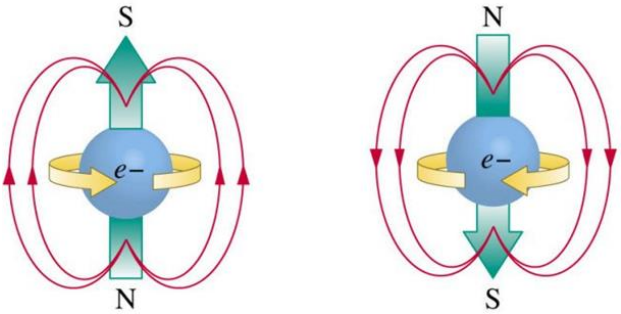


Рис. 2. Спин электрона

Оказалось возможным проверить, сколько времени потребуется электрону, чтобы изменить свой спин (и испустить при этом фотон) - от одной до 20 наносекунд [10]. Это время может служить ориентиром при создании элементов цифровых квантовых сопроцессоров – цифровых квантовых вентилей.

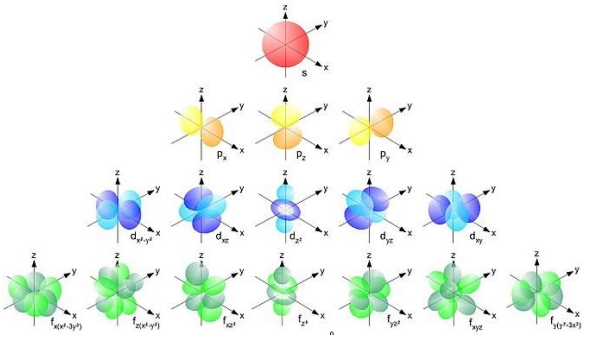


Рис. 3. Форма и расположение в пространстве s-, p-, d- и f-орбиталей

2. Квантовые вентили

Квантовые вентили (элементы квантовой логики) являются базовыми элементами квантового компьютера, которые изменяют состояние кубит по определенным законам в зависимости от сигналов на их входах. Из-за необходимости работы с кубитами квантовые вентили подчиняются квантовой логике.

Квантовые вентили в отличие от классических вентилей всегда обратимы. Изменение состояния кубита определяется цепочкой квантовых вентилей.

Существует небольшой набор квантовых вентилей (которые выполняют условное вращение вектора и преобразование Адамара), который позволяет организовать квантовое преобразование Фурье [7].

Поскольку примитивные операции, которые выполняют квантовые вентили, аналогичны операциям, которые выполняют логические элементы в классических вычислениях, системы, построенные с использованием этого подхода, иногда называются «цифровыми квантовыми компьютерами» [11]. Однако, надо понимать, что в этом случае так называемые «цифровые квантовые компьютеры» все равно являются аналоговыми компьютерами, в отличие от настоящих цифровых квантовых компьютеров, описанных в этой статье и в [1], [2], [12], [13].

Существуют программные средства, позволяющие описывать схемы, состоящие из квантовых вентилей, и моделировать их работу [14].

3. Цифровой квантовый компьютер

Настоящий квантовый компьютер - это аналоговый вероятностный компьютер. Его узлы состоят только из аналоговых вентилей. В них нет элементов памяти. Поэтому не существует квантовых программ – все вычисления, которые производит квантовый компьютер, полностью определяются его схемой. Существуют программные средства, которые позволяют описывать на *C*-подобном языке [14] схемы, состоящие из квантовых вентилей, и моделировать их. Это очень похоже на проектирование аналоговых программируемых схем.

Только классический компьютер, который управляет квантовым компьютером, выполняет реальные программы. Таким образом, на самом деле квантовый компьютер является сопроцессором по отношению к классическому компьютеру. Их взаимодействие иллюстрирует рис. 4 [2].

Многокубитный квантовый сопроцессор состоит из нескольких кубитов, можно представить его как набор однокубитных квантовых сопроцессоров, которые все подключены к одному классическому компьютеру, управляются им (рис. 5) и могут влиять на работу друг друга.

Классический компьютер управляет работой квантового сопроцессора, предоставляет ему входные данные, поток инструкций и проверяет результат его работы.

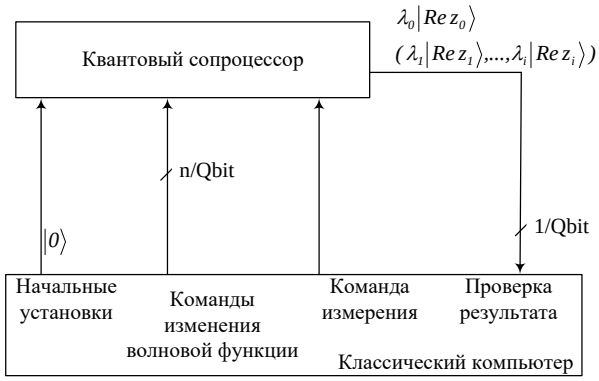


Рис. 4. Классический компьютер с квантовым сопроцессором

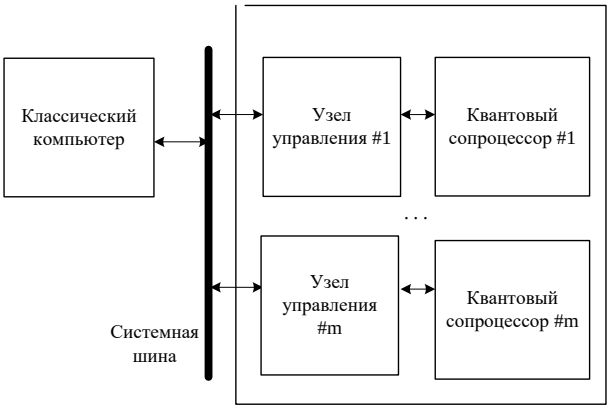


Рис. 5. Классический компьютер с несколькими квантовыми сопроцессорами

Если удалось создать программную модель аналогового квантового компьютера [14], то можно создать и его аппаратную цифровую версию – цифровой квантовый компьютер и реализовать его на ПЛИС ([1], [2], [12], [13]).

Цифровой квантовый сопроцессор состоит из цифровых кубитов. Каждый цифровой кубит – это цифровой автомат, изменение состояния которого описывается так же как и изменение состояния аналогового кубита (такими же формулами).

Цифровой кубит – это цепочка цифровых квантовых вентилей. Каждый цифровой квантовый вентиль – это цифровой автомат, изменение состояния которого описывается так же как и изменение состояния аналогового квантового вентиля (такими же формулами). Основное отличие цифровых кубита и цифрового квантового вентиля от аналоговых состоит в наличии у цифровых памяти (точнее, в возможности ее введения в их схему) и ее использовании для организации вычислений.

Использование цифровых версий квантовых вентилей и кубитов упрощает построение и

работу квантовых сопроцессоров (очень напоминает ситуацию с аналоговым и цифровым телевидением).

Некоторые варианты построения цифровых квантовых компьютеров обсуждаются в [15].

В [16] рассмотрен вариант построения цифрового квантового процессора, где состояния кубитов описываются комплексными числами. Это приводит к усложнению вычислений, к необходимости выполнения умножений в ходе реализации рассмотренных в работе алгоритмов квантового преобразования Фурье и алгоритма поиска Гровера.

Обобщенная функциональная схема однокубитного цифрового квантового сопроцессора приведена на рис. 6 [2].

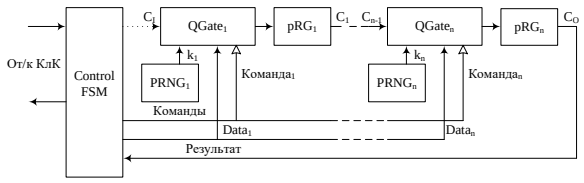


Рис. 6. Обобщенная функциональная схема цифрового квантового сопроцессора

Цифровой квантовый сопроцессор (на 1 кубит) на рис. 6 управляется со стороны классического компьютеры (КлК) и представлен в виде набора конечных автоматов (*FSM*), один из которых является контроллером, а еще один или несколько реализуют функции цифровых квантовых вентилей. Связи между последними могут осуществляться через конвейерные регистры (*pRG*).

Введение в схему цифрового квантового сопроцессора дополнительной программируемой коммутационной матрицы делает его похожим на программируемые логические матрицы (*CPLD* [17], Рис. 7 [2]).

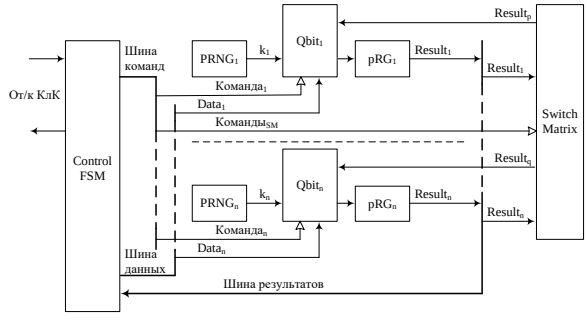


Рис. 7. Обобщенная функциональная схема цифрового квантового сопроцессора с коммутационной матрицей

Входы коммутационной матрицы - это выходы всех цифровых кубитов (*Result*$_{1,\ldots,}$ *Result*$_n$), со стороны классического компьютера матрица может быть запрограммирована таким образом, чтобы на входы любого кубита подавалось состояние любого другого кубита или кубитов (*Result*$_{p,\ \ldots,}$ *Result*$_q$).

Простейшая версия цифрового квантового сопроцессора на ПЛИС имеет только один цифровой универсальный квантовый вентиль, который последовательно программируется со стороны классического компьютера на выполнение операций, которые соответствуют последовательности квантовых вентилей в кубитах. Классический компьютер определяет последовательность всех операций.

Более сложный цифровой квантовый сопроцессор имеет несколько цепочек цифровых квантовых вентилей (на рис. 6 - сопроцессор с одной цепочкой). Передача данных от одного цифрового вентиля к другому может осуществляться через конвейерные регистры (с получением всех преимуществ конвейерных структур).

Классический аналоговый кубит формирует только один выходной бит. Применение цифровых схем позволяет дополнительно получать на выходе цифрового кубита многобитный код, который точно описывает текущеее состояние кубита (рис. 8). Этот код дает возможность классическому компьютеру организовывать на одном цифровом кубите последовательную обработку данных, которые в аналоговом квантовом процессоре обрабатывают несколько кубитов.

Использование дополнительного многобитового выхода классическим компьютером позволяет организовать прерывания квантовых программ и вызывать квантовые процедуры и функции (программы, процедуры, функции и прерывания, которые выполняются и обрабатываются классическим компьютером в процессе управления цифровым квантовым сопроцессором, будем называть квантовыми).

Цифровой кубит как конечный цифровой автомат (т.е. схема с обратной связью) был описан в [2]; в [13] и в данной работе используются цифровые кубиты без обратных связей – цифровые квантовые вентили, но с конвейерными регистром и триггером на выходах (рис. 8).

В [2] была исследована возможность получения правильных вероятностных результатов с помощью цифровых квантовых вентилей, кубитов и цифрового квантового сопроцессора. Способность 4-кубитного цифрового квантового сопроцессора выполнять более сложный алгоритм (квантовое преобразование Фурье), который используется в

алгоритме Шора, аппаратные и временные затраты на его реализацию на ПЛИС были рассмотрены в [13].

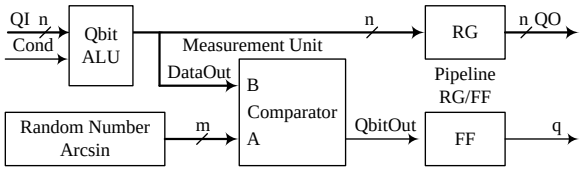


Рис. 8. Цифровой квантовый вентиль

4. Цель работы

В данной работе исследуется работа реализованных на ПЛИС цифровых квантовых сопроцессоров с количеством кубит до 128. Анализируется влияние на параметры сопроцессора разрядности цифрового кубита и цифровых квантовых вентилей, а также аппаратные и временные характеристики различных сопроцессоров.

5. Цифровой кубит для случая действительных амплитуд волновой функции

Для проверки возможности построения цифрового кубита, он был реализован для случая действительных амплитуд волновой функции, которая описывает его поведение. В этом случае поведение одного кубита описывается движением радиус-вектора в единичной окружности (рис. 1). В этой статье (как и в предыдущих работах [2], [12], [13]) для представления положения вектора выбрана полярная система координат (при этом необходимо указать и обрабатывать только одну координату - угол θ (рис. 9), в отличие от декартовой системы координат, где необходимо обрабатывать две координаты x и y). Также в полярной системе координат без выполнения умножения выполняется существенная для квантового преобразования Фурье операция фазового сдвига.

Существует несколько подходов к проектированию квантовых компьютеров на ПЛИС ([15], [18]). Особенностью [2], [12], [13] и данной статьи является реализация цифрового кубита с использованием полярных координат для представления положения вектора на единичной окружности. Для реализации вероятностных функций используется генератор псевдослучайных кодов с периодом 2^{32}-1 [19], [20]. На выход генератора случайных кодов добавлен функциональный преобразователь [12] (преобразователь $D = arcsin\sqrt{A}$ на рис. 10).

Функциональные преобразователи могут быть созданы по известным методам [21], [22].

Коды, используемые для представления угловых положений вектора в единичном круге, показаны на рис. 9. При этом угол $\pi/2$ имеет двоичный код 10…0, угол $\pi/4$ - двоичный код 010…0 и .т.д. Угол 0 радиан имеет код 0…0.

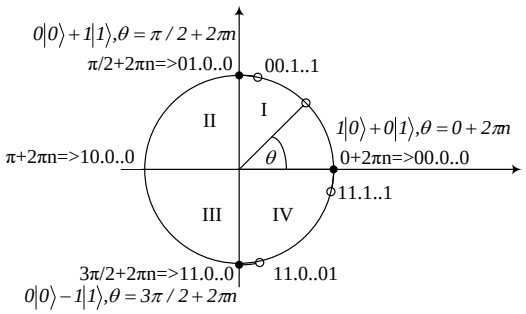


Рис. 9. Вектор в единичном круге

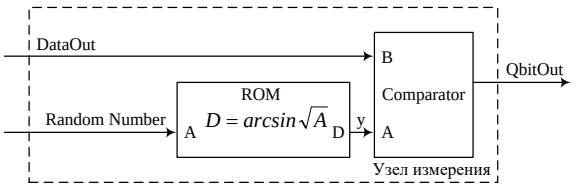


Рис. 10. Узел измерения с функциональным преобразователем

Цифровой кубит без обратных связей – цифровой квантовый вентиль, которымм можно управлять с помощью классического компьютера, представляет собой одну ступеньку конвейера (рис. 8) с узлом измерения (компаратором), конвейерными регистром и конвейерным триггером на выходе.

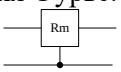
Цифровой квантовый вентиль рис. 8 имеет многобитовые вход *QI* и выход *QO*, однобитовый выход *q* и вход для условий *Cond*, который используется для выполнения условных команд, таких как условный сдвиг фазы в квантовом преобразовании Фурье. Дополнительный и опционный многобитовый выход (*QO* на рис. 8) позволяет организовывать прерывания программ управления цифровым кубитом, выполняемых классическим компьютером. Этот выход позволяет считывать точное внутреннее состояние *S0* цифрового кубита, сохранять его в памяти классического компьютера, загружать новое состояние *S1* через вход *QI*, выполнять в кубите последовательность инструкций, начиная с этого состояния *S1*, считывать точное состояние *S2*, в котором кубит оказывается после их выполнения, сохранять *S2* в памяти классического компьютера, затем читать из памяти классического компьютера ранее сохраненное предыдущее состояние *S0* кубита, загружать его в кубит и продолжать выполнение ранее прерванной последовательности

инструкций начиная с этого состояния *S0*. Аналогично можно последовательно выполнять на одном цифровом кубите операции над большим количеством кубитов.

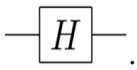
6. Основные операции, необходимые для выполнения квантового преобразования Фурье

Набор инструкций, которые выполняет цифровой кубит, должен быть универсальным – должен составлять функционально полную систему, чтобы используя одну или несколько из них, можно было воспроизвести поведение любого квантового вентиля. Квантовые вентили [7] выполняют унитарные операции, которые не меняют величину вектора на сфере Блоха или в одиночном круге; они только перемещают конец вектора по сфере или по кругу.

Только 2 из таких операций требуются для выполнения квантового преобразования Фурье:

• контролируемый сдвиг фазы R_m 

;

• преобразование Адамара H 

.

Эти две операции описываются в матричном виде

$$R_m = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{2\pi i/2^m} \end{pmatrix}, \quad H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

7. Алгоритм Шора

Задача разложения на множители числа M сводится к задаче определения периода r функции $y = 2^x mod M$. Показано, что наименьший общий делитель НОД($2^{r/2}$+1,M) может быть делителем числа M.

Квантовый процессор позволяет для каждого предполагаемого значения T_j из какого-то диапазона значений определить вероятность p_j того, что оно является периодом функции $y = 2^x mod M$. Поскольку период T является обратной функцией частоты F ($T = 1/F$), проще определить p_j анализируя частоты F_j. Для этого в алгоритме Шора [3] используется квантовое преобразование Фурье [7], которое может выполнять предлагаемый цифровой квантовый сопроцессор.

8. Квантовое преобразование Фурье

В квантовых вычислениях квантовое преобразование Фурье (*QFT*) [7] представляет собой линейное преобразование кубитов и является квантовым аналогом обратного дискретного преобразования Фурье. Квантовое преобразование Фурье является частью многих квантовых алгоритмов, в частности алгоритма факторизации (разложения на множители) Шора [3].

Квантовое преобразование Фурье определяется [23] как $\sum_j a_j |j\rangle \to \sum_k \tilde{a}_k |k\rangle$, где $\tilde{a}_k \equiv \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i j / N_{a_j}}$. Квантовыми вентилями, используемыми в схеме *QFT* (рис. 11, рис. 12), являются вентиль Адамара (H) и вентиль управляемого смещения фазы (R_m). При этом смещения фазы равны $\pi/2^{m-1}$.

Задача дискретного квантового преобразования Фурье состоит в определении частоты F изменения измеренных состояний квантового процессора (точнее, определение вероятностей того, что отдельные частоты из частотного спектра будут равны искомой частоте). В алгоритме Шора результат используется для определения периода изменения состояний T, поскольку $T = 1 / F$.

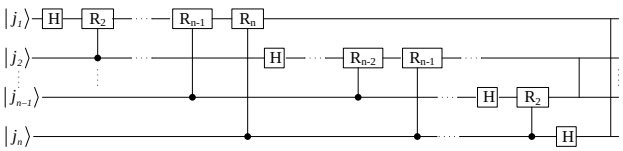


Рис. 11. Общая схема квантового преобразования Фурье [23]

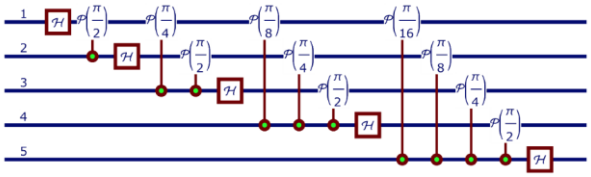


Рис. 12. Упрощенная схема квантового преобразования Фурье для 5 кубит [23]

Наглядный пример квантового преобразования Фурье, примененного к схеме, состоящей из 3-х кубит, приведен в [11].

9. Анализ результатов квантового преобразования Фурье, выполняемого цифровым квантовым сопроцессором

Для исследования различных вариантов цифровых квантовых сопроцессоров был разработан соответствующий генератор их *VHDL*-описаний (ядер, *IP Cores*). При синтезе каждого ядра задается:

количество кубит QB_Number;

разрядность кода состояния кубита Width;

разрядность генератора псевдослучайных кодов.

Пример графического символа, полученного из сгенерированного VHDL-описания 128-кубитного квантового сопроцессора для выполнения квантового преобразования Фурье, представлен на рис. 13.

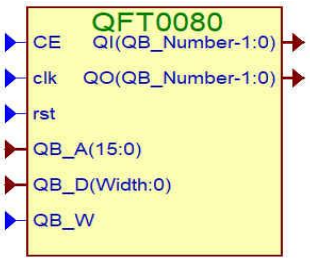


Рис. 13. Графический символ элемента квантового преобразования Фурье для 128 кубит

Символ имеет входы:

QB_D – данные для установки начального состояния каждого кубита;

QB_A – адрес кубита в адресном поле классического компьютера;

QB_W – сигнал записи от классического компьютера;

CE – разрешение работы от классического компьютера;

clk, rst – системные сигналы синхроимпульсов и сброса.

На выходе сопроцессора формируется коды его входного состояния QI (до преобразования) и его выходного состояния QO (после преобразования).

Подробная схема реализованного на ПЛИС 4-кубитного цифрового квантового сопроцессора, который выполняет дискретное преобразование Фурье была описана в [13]. На рис. 14 показана структура 6-кубитного квантового сопроцессора, которая позволяет сделать обобщающие выводы про особенности выполнения квантового преобразования Фурье на цифровом квантовом сопроцессоре с произвольным количеством кубитов. Схема соответствует рис. 11 и рис. 12.

Схема рис. 14 имеет 6 рядов элементов. Каждая строка соответствует преобразованию одного аналогового или одного цифрового кубита.

Цифровой квантовый сопроцессор на рис. 14 состоит из трех типов цифровых квантовых вентилей *QG* (рис. 10).

В цифровых квантовых элементах типа *Load* арифметико-логический узел *ALU* передает входной код *QI* в регистр конвейера без преобразования (трансляция существующего кода, входного или промежуточного при вычислениях).

В цифровых квантовых элементах типа *Had ALU* выполняет над входным кодом преобразования Адамара и записывает результат в конвейерный регистр.

В цифровых квантовых элементах типа *Add Const ALU* вычисляет сумму *QI* + *Const/2*, если *Cond* = 1 (где *Const* - π/2, или π/4, или π/8) и записывает результат в выходной конвейерный регистр. Для упрощения объяснений предположено, что состояние квантового вентиля (текущая фаза вектора) кодируется тремя битами. Следовательно, и условное смещение фазы (то есть, добавление к существующей фазе какого-то смещения) должно выполняться на 3-битном АЛУ, а следовательно и смещения должны быть 3-битными. Например, смещение π/2 может кодироваться как π/2 = 100, π/4 = 010, π/8 = 001.

Меньшие смещения при таком способе кодировки становятся равным 0 и треугольные структура рис. 11 и рис. 12 превращается в пирамидальную рис. 14, а при значительном увеличении количества квантовых вентилей – практически в прямоугольную. При этом каждый ряд цифровых квантовых вентилей (цифровой кубит) структуры рис. 14 связан с ближайшими нижними рядами (цифровыми кубитами) только тремя связями (количество связей совпадает с разрядностью кода состояния кубита).

Прямоугольная структура цифрового квантового сопроцессора указывает на то, что:

временная сложность квантового преобразования Фурье не зависит от количества кубит в цифровом квантовом сопроцессоре, а значит не зависит от величины (разрядности) числа, которое раскладывается на множители по алгоритму Шора, и является константой;

аппаратная сложность квантового преобразования Фурье линейно зависит от количества *n* кубит в цифровом квантовом сопроцессоре, а значит и от разрядности числа, которое раскладывается на множители по алгоритму Шора, т. е. равна *O(n)*.

В теории сложности, *PP* является классом проблем, решаемых вероятностными машинами Тьюринга за полиномиальное время, с вероятностью ошибки менее ½ [24]. Аббревиатура *PP* обозначает «вероятностный полиномиальный по времени». Таким образом аппаратная сложность квантового преобразования Фурье является вероятностной полиномиальной по времени.

Благодаря наличию памяти в структуре цифровых кубитов число цифровых квантовых вентилей в схеме сопроцессора может быть уменьшено до одного ряда, или одного столбца, или даже до одного цифрового квантового вентиля (одного цифрового кубита). В этом

случае АЛУ в каждом цифровом квантовом вентиле будет выполнять не одну, а несколько операций, задаваемых со стороны классического компьютера. При этом схема управления на рис. 14 будет более сложной.

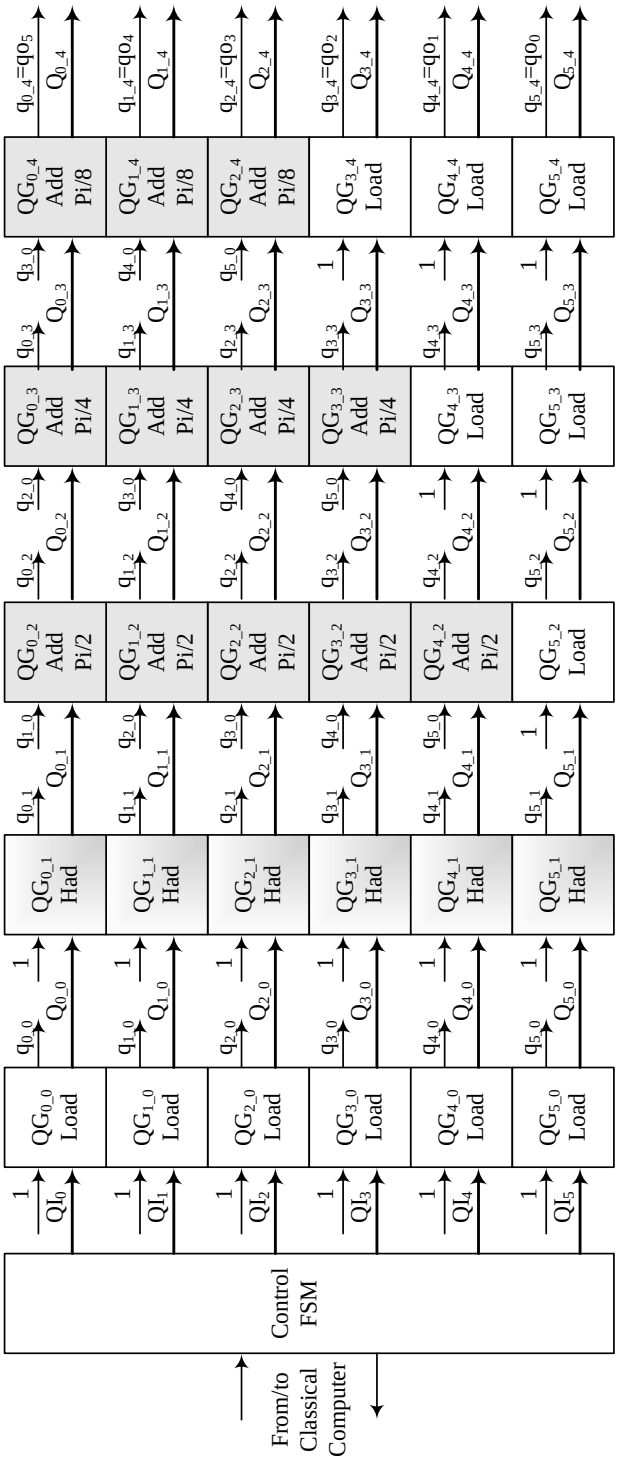


Рис. 14. Детальная схема квантового преобразования Фурье для 6 кубит [13]

Результаты исследования представлены в виде графиков. График (*a*) показывают спектр состояний на входе 4-кубитового узла для квантового преобразования Фурье, а график (*b*) - спектр его выходных состояний. Состояния кубитов до преобразования, которые определяют спектр входных состояний (в виде углов, определяющих положение векторов в единичной окружности), задаются до начала исследования и в ходе исследования не изменяются. Вероятность нахождения кубитов до преобразования в одном из 16 состояний вычисляется. Например, вероятность того, что измерение входного состояния даст его значение равное $|13\rangle = |q_3 q_2 q_1 q_0\rangle = |1101\rangle$ будет $P_{13} = cos^2\,\theta_3 \cdot cos^2\,\theta_2 \cdot sin^2\,\theta_1 \cdot cos^2\,\theta_0$, где θ_j - это угол, который определяет положение вектора *j*-го кубита q_j.

Для исследования выбирается ситуация, когда один вектор расположен под углом 0°, а все остальные находятся под углом 45°. Тогда в спектре присутствует только одна частота. Для примера на рис. 15 показаны результаты исследования, когда $\theta_3 = \theta_2 = \theta_1 = 45°, \theta_0 = 0°$, т. е., для N = 4 кубитов, количество возможных состояний $S = 2^N$=16 (0 … 15), а искомая частота $F = 2^{N-1}$=8.

На рис. 9.2 обозначено:

Probability – Вероятность:

Input states – Входные состояния;

Input states spectrum – Спектр входных состояний;

Output states – Выходные состояния;

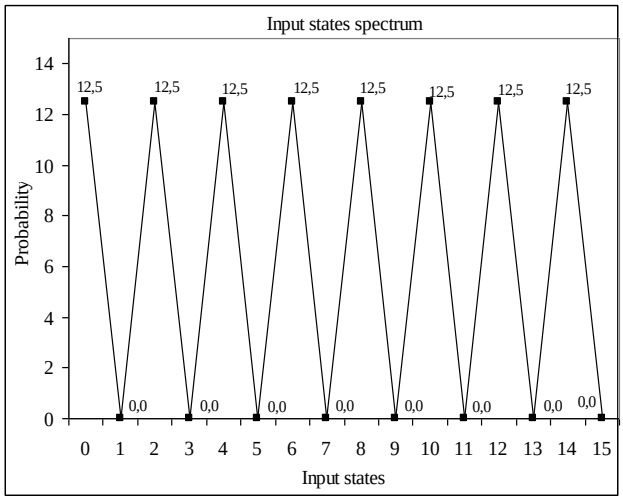
Output states spectrum – Спектр выходных состояний.

Как видно из рис. 15 (а), спектр входных состояний является периодическим и состоит из 8 периодов изменения вероятностей.

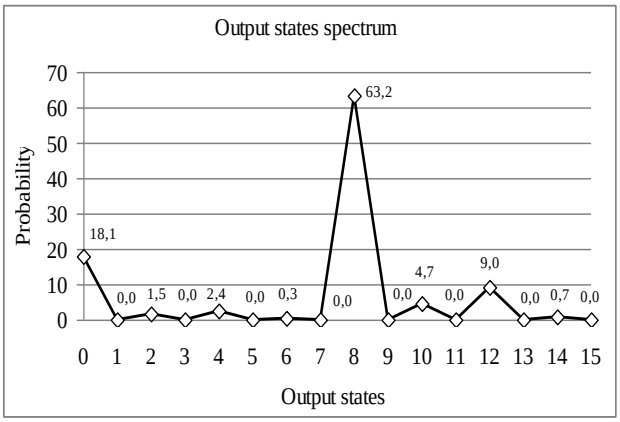
Квантовый сопроцессор отметит это явление после квантового преобразования Фурье и выдаст истинный результат 8 (как число периодов колебаний) с вероятностью 63,2%, или даст ложный результат 0 с меньшей вероятностью 18,1%, или даст другие ложные результаты с еще более низкой вероятностью.

Задача классического компьютера - проверить каждый из результатов квантового сопроцессора и использовать его для завершения алгоритма Шора.

Спектр выходных состояний определялся путем измерения состояний цифрового квантового сопроцессора после каждого из 10000 идентичных экспериментов с последующим вычислением частоты возникновения (вероятности) каждого выходного состояния.



a



b

Рис. 15. Спектры входных (*a*) и выходных (*b*) состояний для случая $\theta_3 = \theta_2 = \theta_1 = 45°, \theta_0 = 0°$

Как было отмечено в [13] 4-кубитный цифровой квантовый сопроцессор при проведении описанных экспериментов в целом правильно определяет число периодов входных состояний, то есть он правильно выполняет квантовое преобразование Фурье. Это указывает на то, что предложенные цифровые квантовые вентили и кубиты, из которых он состоит, также работают надлежащим образом; они могут быть использованы для создания других цифровых квантовых сопроцессоров.

10. Исследование цифрового квантового сопроцессора с количеством кубит до 128

Результаты имплементации синтезированных с использование описанного генератора ядер цифровых квантовых сопроцессоров для выполнения квантового преобразования Фурье сведены в таблицу 1.

Таблица 1

Результаты имплементации цифровых квантовых сопроцессоров

N	*W*	*%H*	*T, ns*	*LUTS*	*RAM*	*%R*	*7z*
128	3		2.353	32352		14	045
64	3	1,5	4.289	16043		15	010
32	3	7	3.874	7983			010
16	3	22	3.606	4000			010
10	3	33	3.976	2522			010
10	5	24	4.180	4307		24	010
10	7	22	5.137	3754			010
10	9	20	5.281	3650	20	16	010

В таблице 1 обозначено:

N – количество кубит в цифровом квантовом сопроцессоре;

W – количество бит в коде состояния кубита (в коде фазы его вектора);

%*H* – процент правильных решений, найденных цифровым квантовым сопроцессором для ситуации, когда искомая частота $F = 2^{N-1}$;

T, ns – период тактовой частоты работы цифрового квантового сопроцессора;

LUTS – количество задействованных в ПЛИС логических элементов;

RAM – количество задействованных в ПЛИС 18-битных блоков памяти;

%*R* – процент использованных ресурсов ПЛИС;

7*z* – тип используемой ПЛИС *Zynq*-7000 (*Xilinx*, 7Z010 или 7z045, обе -3, [25]).

Дальнейшее заполнение таблицы 1 не выполнялось из-за увеличения времени моделирования.

Результаты имплементации подтверждают практически неизменное время вычислений и линейную аппаратную сложность.

Время вычисления соизмеримо с временем изменения спина электрона.

Поскольку для реализации 128-кубитного цифрового квантового сопроцессора необходимо всего 14 % ресурсов ПЛИС, то можно оценить что 1024-кубитный сопроцессора потребует примерно 100 % ресурсов кристалла ПЛИС (в 8 раз больше). Это позволяет рассчитывать на появление килокубитных (*K*-кубитных) цифровых квантовых сопроцессоров на одном кристалле ПЛИС.

Результаты показывают увеличение результативности работы цифровых квантовых сопроцессоров при уменьшении разрядности кодов их состояний. Например, 10-кубитный сопроцессор с уменьшением разрядности его кубитов в последовательности 9, 7, 5, 3

соответственно увеличивает процент правильных решений (20 %, 22 %, 24 % и 33 % соответственно).

С увеличением количества кубит уменьшается результативность работы цифрового квантового сопроцессора, при 64 кубитах она составляет 1,5 %, уменьшение нелинейное.

Процент правильных решений позволяет оценить среднее количество экспериментов N_E, которые необходимо провести до получения первого правильного решения $N_E = \lceil 1/\%H \rceil$. Время проведения каждого эксперимента по схеме рис. 14 $T_E=5T$ (в каждой строке находится 5 цифровых квантовых вентилей). Результаты оценки ожидаемого времени выполнения квантового преобразования Фурье цифровым квантовым сопроцессором приведены в таблице 2.

Таблица 2

Время квантового преобразования Фурье, нс

N	*W*	*%H*	*T, ns*	*NE*	*TE, ns*
128	3		2.353		
64	3	1,5	4.289	67	287
32	3	7	3.874	15	58
16	3	22	3.606	5	18
10	3	33	3.976	4	16
10	5	24	4.180	5	21
10	7	22	5.137	5	26
10	9	20	5.281	5	26

Выводы

Приведены результаты выполнения реализованным на одной ПЛИС многокубитным цифровым квантовым сопроцессором квантового преобразования Фурье, которое является основной квантовой операцией алгоритма Шора. Для этого были созданы генератор ядер (моделей) многокубитного квантового сопроцессора, промоделирована его работа, после чего модель была имплементирована в ПЛИС и были определены ее временные и аппаратные характеристики.

Установлено что:

временная сложность квантового преобразования Фурье не зависит от количества кубит в цифровом квантовом сопроцессоре и является константой;

аппаратная сложность квантового преобразования Фурье линейно зависит от количества кубит в цифровом квантовом сопроцессоре.

Список использованной литературы

1. Глухов, Валерий. Квантовый компьютер как вероятностный компьютер [Текст] / Валерий Глухов // Шоста міжнародна наукова конференція «Моделювання-2018» : Збірка праць конференції (ІПМЕ ім. Г.Є. Пухова НАН України, Київ, Україна, 12-14 вересня 2018), Київ : ВД «Академперіодика» НАН України, 2018. - с. 111–114.

2. Hlukhov, Valerii. FPGA-based Digital Quantum Coprocessor [Text] / Valerii Hlukhov, Bohdan Havano // Advances in Cyber-Physical Systems. - Lviv, 2018. - Volume 3. Number 2. - Pp. 12–31.

3. Shor, Peter W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer [Text] // Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, Nov. 20–22, 1994, IEEE Computer Society Press. Pp. 124–134.

4. Applying Moore's Law to Quantum Qubits [Electronic Resource] / Quantum Computing Report: [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : https://quantumcomputingreport.com/our-take/applying-moores-law-to-quantum-qubits.

5. Al-Ta'ani, Ola. Implementation and Analysis of Quantum Fourier Transform in Image Processing [Electronic Resource] / Ola Al-Ta'ani, Ali Mohammad Alqudah, Manal Al-bzoor / ResearchGate: [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : https://www.researchgate.net/publication/331674710_Implementation_and_Analysis_of_Quantum_Fourier_Transform_in_Image_Processing.

6. Hlukhov, V. Hardware components for post-quantum elliptic curves cryptography [Electronic Resource] / R. Elias, V. Hlukhov, M. Rahma, I. Zholubak // Proceedings of the International Conference Advanced Computer Information Technologies (Ceske Budejovice, Czech Republic, June 1-3, 2018) / ACIT 2018: [офіц. веб-сайт]. – Електрон. дані. – 2018. – Режим доступу : http://ceur-ws.org/Vol-2300/Paper57.pdf.

7. Nielsen, M. Quantum Computation and Quantum Information Theory [Text] / M. Nielsen, I. Chuang. - Cambridge Press, 2000.

8. Спин электрона и тонкая структура спектров [Электронный ресурс] / Образовательный портал НИЯУ МИФИ: [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : http://online.mephi.ru/courses/physics/atomic_physics/data/course/5/5.4.html.

9. Орбиталь [Электронный ресурс] / Википедия: [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : https://ru.wikipedia.org/wiki/Орбиталь.

10. Physicists control the flip of electron spin [Electronic Resource] / Phys.org: [офіц. веб-сайт]. – Електрон. дані. – 2005. – Режим доступу : https://phys.org/news/2005-05-physicists-flip-electron.html.

11. Quantum Computing: Progress and Prospects [Text] / Emily Grumbling and Mark Horowitz, Editors. - Washington, DC : The National Academies Press, 2019. – 272 p.

12. Hlukhov, V. Principles of Digital Quantum Coprocessor Based on a FPGA, which Operates under the Control of a Classical Computer [Electronic Resource] / Valeriy Hlukhov, Bohdan Havano / IEEE Xplore : [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : https://ieeexplore.ieee.org/document/8779932.

13. Hlukhov, Valerii. Implementing Quantum Fourier Transform in a Digital Quantum Coprocessor [Text] / Valerii Hlukhov // Advances in Cyber-Physical Systems. – Lviv, 2019. - Volume 4. Number 1. - Pp. 6–13.

14. Welcome to the Microsoft Quantum Development Kit Preview [Electronic Resource] / Microsoft Docs: [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : https://docs.microsoft.com/ru-ru/quantum/?view=qsharp-preview.

15. Khalil-Hani, M. An Accurate FPGA-Based Hardware Emulation on Quantum Fourier Transform [Text] / M. Khalil-Hani, Y. H. Lee, M. N. Marsono // Proceedings of the 13th Australasian Symposium on Parallel and Distributed Computing (AusPDC 2015, 27 - 30 January 2015), Sydney, Australia: 2015. - Pp. 23–30.

16. Lee, Y. H. An FPGA-Based Quantum Computing Emulation Framework Based on Serial-Parallel Architecture [Electronic Resource] / Y. H. Lee, M. Khalil-Hani, M. N. Marsono / Hindawi: [офіц. веб-сайт]. – Електрон. дані. – 2016. – Режим доступу : https://www.hindawi.com/journals/ijrc/2016/5718124/.

17. CPLD [Electronic Resource] / Xilinx: [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : https://www.xilinx.com/products/silicon-devices/cpld/cpld.html.

18. Gushanskiy, S. M. Simulation of Quantum Computing using Hardware Cores [Electronic Resource] / S. M. Gushanskiy, V. A. Pereverzev // Nauchnyy zhurnal KubGAU: [офіц. веб-сайт]. – Електрон. дані. – 2016. – Режим доступу : http://ej.kubagro.ru/2016/09/pdf/37.pdf.

19. LFSR-Random number generator. Overview [Electronic Resource] / OpenCores: [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : https://opencores.org/projects/lfsr_randgen.

20. Pseudo Random Number Generators as synthesizable VHDL code [Electronic Resource] / GitHub: [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : https://github.com/jorisvr/vhdl_prng.

21. Попов Б. А., Теслер Г. С. Вычисление функций на ЭВМ. Справочник [Текст]. Киев: Наук. думка, 1984.

22. Аристов, В. В. Интегро-алгоритмические вычисления [Текст]. "Наук. думка", 1980 - Всего страниц: 189.

23. Hui, Jonathan. QC — Quantum Fourier Transform [Electronic Resource] / Medium: [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : https://medium.com/@jonathan_hui/qc-quantum-fourier-transform-45436f90a43.

24. Gill, John (1977). Computational Complexity of Probabilistic Turing Machines [Text] // SIAM Journal on Computing. - Philadelphia, PA, 1977. - 6(4): - 675–695. doi:10.1137/0206049.

25. DS190 (v1.11.1) Zynq-7000 SoC Data Sheet: Overview [Electronic Resource] / Xilinx: [офіц. веб-сайт]. – Електрон. дані. – 2019. – Режим доступу : https://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf.

References

1. Hlukhov, V. (2018), "Quantum computer as a probabilistic computer", Sixth International Scientific Conference "Modeling 2018", Conference Proceedings ["Kvantovyj komp'juter kak verojatnostnyj komp'juter", Shosta mizhnarodna naukova konferentsiia «Modeliuvannia-2018». Zbirka prats konferentsii], Pukhov IPEE NAS of Ukraine, Kyiv, Ukraine, September 12-14, 2018, pp. 111–114 (In Russian).

2. Hlukhov, V., Havano, B., (2018), "FPGA-based Digital Quantum Coprocessor", Advances in Cyber-Physical Systems, Volume 3, Number 2, Lviv, 2018, Pp. 12–31.

3. Shor, Peter W. (1994), "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer", Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, Nov. 20–22, 1994, IEEE Computer Society Press. Pp. 124–134.

4. "Applying Moore's Law to Quantum Qubits" (2019), available at:

https://quantumcomputingreport.com/our-take/applying-moores-law-to-quantum-qubits.

5. Al-Ta'ani, O., Alqudah, A.M., Al-bzoor, M. (2019), "Implementation and Analysis of Quantum Fourier Transform in Image Processing", available at: https://www.researchgate.net/publication/331674710_Implementation_and_Analysis_of_Quantum_Fourier_Transform_in_Image_Processing.

6. Hlukhov, V., Elias, R., Zholubak I. (2018), "Hardware components for post-quantum elliptic curves cryptography", Proceedings of the International Conference Advanced Computer Information Technologies, Ceske Budejovice, Czech Republic, June 1-3, 2018, available at: http://ceur-ws.org/Vol-2300/Paper57.pdf.

7. Nielsen, M., Chuang I. (2000), "Quantum Computation and Quantum Information Theory", Cambridge Press.

8. "Spin of an electron and fine structure of spectra" [Spin jelektrona i tonkaja struktura spektrov], available at: http://online.mephi.ru/courses/physics/atomic_physics/data/course/5/5.4.html (In Russian).

9. "Orbital" [Orbital'], available at: https://ru.wikipedia.org/wiki/Орбиталь (In Russian).

10. "Physicists control the flip of electron spin" (2005), available at: https://phys.org/news/2005-05-physicists-flip-electron.html (In Russian).

11. Grumbling, E., Horowitz, M. (2019), Quantum "Computing: Progress and Prospects", Washington, DC : The National Academies Press, 272 p.

12. Hlukhov, V., Havano, B. (2019), "Principles of Digital Quantum Coprocessor Based on a FPGA, which Operates under the Control of a Classical Computer", available at: https://ieeexplore.ieee.org/document/8779932.

13. Hlukhov, V. (2019), "Implementing Quantum Fourier Transform in a Digital Quantum Coprocessor", Advances in Cyber-Physical Systems, Lviv, Volume 4, Number 1, 6–13.

14. "Welcome to the Microsoft Quantum Development Kit Preview", available at: https://docs.microsoft.com/ru-ru/quantum/?view=qsharp-preview.

15. Khalil-Hani, M., Lee, Y. H., Marsono, M. N. (2015), "An Accurate FPGA-Based Hardware Emulation on Quantum Fourier Transform", Proceedings of the 13th Australasian Symposium on Parallel and Distributed Computing, 27 - 30 January 2015, Sydney, Australia, 23–30.

16. Lee, Y. H., Khalil-Hani, M., Marsono, M. N. (2016), "An FPGA-Based Quantum Computing Emulation Framework Based on Serial-Parallel Architecture", available at: https://www.hindawi.com/journals/ijrc/2016/5718124/.

17. "CPLD", available at: https://www.xilinx.com/products/silicon-devices/cpld/cpld.html.

18. Gushanskiy, S. M., Pereverzev, V. A. (2016), "Simulation of Quantum Computing using Hardware Cores", available at: http://ej.kubagro.ru/2016/09/pdf/37.pdf (In Russian).

19. "LFSR-Random number generator. Overview", available at: https://opencores.org/projects/lfsr_randgen.

20. "Pseudo Random Number Generators as synthesizable VHDL code", available at: https://github.com/jorisvr/vhdl_prng.

21. Popov, B. A.., Tesler, G. S. (1984), "Calculation of functions on a computer. Directory." [Vychislenie funkcij na EVM. Spravochnik], Kiev, Nauk. Dumka, 59 p. (In Russian).

22. Aristov, V. V. (1980), "Integro-Algorithmic Computing" [Integro-algoritmicheskiye vychisleniya], Kiev, Nauk. Dumka, 189 p. (In Russian).

23. Hui, J. "QC — Quantum Fourier Transform", available at: https://medium.com/@jonathan_hui/qc-quantum-fourier-transform-45436f90a43.

24. Gill, J. (1977), "Computational Complexity of Probabilistic Turing Machines', SIAM Journal on Computing, Philadelphia, PA, 6(4), 675–695, doi:10.1137/0206049.

25. 'DS190 (v1.11.1) Zynq-7000 SoC Data Sheet: Overview", available at: https://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf.

FPGA-BASED K-QUBIT DIGITAL QUANTUM COPROCESSOR

V. S. Hlukhov

Lviv Polytechnic National University

Abstract*. It is shown that true quantum computers are analog and probabilistic computers and, in fact, they play the role of coprocessors in relation to classical computers. A digital quantum coprocessor consists of digital qubits. Each digital qubit is a finite state machine, the state changes of which is described in the same way as the state changes of an analog qubit. A digital qubit is a chain of digital quantum gates. Each*

digital quantum gate is digital unit, the state changes of which is described in the same way as the state changes of an analog quantum gate. The main difference between digital qubit and digital quantum gate from analog ones is the presence of memory (more precisely, the possibility of its introduction into their circuit) and its use for organizing calculations. A digital qubit without feedbacks - a digital quantum gate that can be controlled using a classic computer, is a single step of a pipeline with a measurement unit (comparator), a pipeline register and a pipeline trigger at the output.

The quantum Fourier transform (QFT) is part of many quantum algorithms, in particular the Shor's factorization algorithm. The structures of digital quantum gates, digital qubits, and a digital quantum coprocessor capable of performing the QFT are presented. The results of the QFT of a multi-qubit digital quantum coprocessor implemented on one FPGA are presented. For this, a IP core generator of multi-qubit quantum coprocessors was created, cores work was simulated. Models were implemented in FPGAs and their time and hardware characteristics were determined. The execution time of one quantum Fourier transform does not depend on the number of qubits in the digital quantum coprocessor and is commensurate with the time of change of the electron spin. The hardware complexity linearly depends on the number of qubits. With an increase in the number of qubits, the rate of true results of QFT decreases, at 64 qubits it is 1.5%, the decrease is nonlinear.

***Keywords**: digital quantum coprocessor, digital qubit, quantum Fourier transform.*

К-КУБІТНИЙ ЦИФРОВИЙ КВАНТОВИЙ КОПРОЦЕСОР НА ПЛІС

В. С. Глухов

Національний університет «Львівська політехніка»

***Анотація**. Показано, що справжні квантові комп'ютери - це аналогові та ймовірнісні комп'ютери і, власне, вони відіграють роль копроцесорів по відношенню до класичних комп'ютерів. Цифровий квантовий копроцесор складається з цифрових кубітів. Кожен цифровий кубіт - це цифровий автомат, зміни стану якого описуються так само, як і зміни стану аналогового кубіта. Цифровий кубіт - це ланцюг цифрових квантових вентилів. Кожен цифровий квантовий вентиль є цифровим вузлом, зміна стану якого описується так само, як і зміни стану аналогового квантового вентиля. Основна відмінність цифрового кубіту та цифрових квантових вентилів від аналогових - це наявність пам'яті (точніше, можливості її введення в їх схеми) та використання її для організації обчислень. Цифровий кубіт без зворотних зв'язків - це цифровий квантовий вентиль, яким можна керувати за допомогою класичного комп'ютера, він являє собою одну сходинку конвеєра з вузлом вимірювання (компаратором), конвеєрними регістром та тригером на виході.*

Квантове перетворення Фур'є є частиною багатьох квантових алгоритмів, зокрема алгоритму факторизації Шора. Представлено структури цифрових квантових вентилів, цифрових кубітів та цифрового квантового копроцесора, здатного виконувати квантове перетворення Фур'є. Представлено результати квантового перетворення Фур'є, що виконувалося багатокубітним цифровим квантовим копроцесором, реалізованим на одній ПЛІС. Для цього було створено генератор ядер (VHDL-описів) багатокубітних квантових копроцесорів, змодельовано роботу ядер. Моделі були реалізовано в ПЛІС, і було визначено їхні часові та апаратні характеристики. Час виконання одного квантового перетворення Фур'є не залежить від кількості кубітів у цифровому квантовому копроцесорі і є співмірним з часом зміни спіна електрона. Апаратна складність лінійно залежить від кількості кубітів. Із збільшенням кількості кубітів частка правильних результатів квантового перетворення Фур'є зменшується, для 64 кубітів вона складає 1,5 %, зменшення - нелінійне.

***Ключові слова:** цифровий квантовий копроцесор, цифровий кубіт, квантове перетворення Фур'є.*

Получено 20.09.2019

Глухов Валерий Сергеевич, доктор технических наук, профессор, профессор кафедры электронных вычислительных машин Национального университета «Львовская политехника», ул. С. Бандеры, 12, Львов, Украина, E-mail: glukhov@polynet.lviv.ua , м/т.: +38-063-75-72-330.

Valeriy Hlukhov, Dr. of Science, Professor, Professor of the Department of Computer Engineering, Lviv Polytechnic National University, S. Bandera Str., 12, Lviv, Ukraine, E-mail: glukhov@polynet.lviv.ua, м/т.: +38-063-75-72-330.

ORCID ID:0000-0002-0542-7447