

## АРХИТЕКТУРА, ПРОЦЕССОР И РАБОТА КВАНТОВОГО КОМПЬЮТЕРА

А.К. Гуц

Краткое изложение устройства квантового компьютера и организации вычислений на нем.

Теория квантового компьютера основывается на *квантовой механике*, созданной в 20-е годы XXI века австрийцем Эрвином Шрёдингером и немцем Вернером Гейзенбергом.

Идея использовать квантовую механику для построения квантового компьютера независимо высказана россиянином Юрием Маниным [4] и американцем Ричардом Фейнманом [5, 6].

### 1. Архитектура квантового компьютера

Квантовый компьютер имеет архитектуру аналогичную классическому компьютеру. Он состоит из:

- регистров памяти,
- процессора, построенного из логических элементов и производящего вычисления,
- устройства ввода информации,
- устройства вывода полученной в ходе вычислений информации.

### 2. Регистры

Память компьютера разбита на регистры. Регистры состоят из некоторого количества разрядов. Регистр из  $m$  разрядов изобразим как



Квадратик изображает разряд

## 2.1. Бит и классический регистр

Классический разряд  $\square$  хранит единицу информации – *бит* информации – 0 или 1.

Запишем разряд в символическом виде

$$|n_k\rangle, \quad n_k = 0, 1.$$

Тогда классический регистр можно представить как

$$|n_{m-1}n_{m-2}\dots n_0\rangle. \quad (1)$$

Для технической реализации бита используются разные физические устройства.

**Пример 1.** Высокий потенциал в точке схемы – 1, низкий – 0.

**Пример 2.** Ферромагнитное колечко намагничено в одном направлении – 1, в другом – 0.

### Состояние классического регистра в момент времени $t$

Разряд классического регистра находится только в одном из двух возможных состояний –  $|0\rangle$  или  $|1\rangle$ .

Поэтому состояние регистра – это

$$|n_{m-1}n_{m-2}\dots n_0\rangle.$$

Например, состояние

$$|\underbrace{01001011100011\dots}_m\rangle.$$

## 2.2. Кубит и квантовый регистр

Квантовый разряд  $\square$  хранит единицу информации – квантовый бит, или *кубит* информации – 0 или 1.

Квантовый разряд в символическом виде выглядит так же, как классический:

$$|n_k\rangle, \quad n_k = 0, 1.$$

И поэтому квантовый регистр представляется в виде

$$|n_{m-1}n_{m-2}\dots n_0\rangle. \tag{2}$$

Для технической реализации кубита предлагаются разные физические устройства, основой которых является любая двухуровневая (квантовомеханическая) система (спин, фотон, атом, молекула, ион).

**Пример 1.** Проекция спина атома (+1) принимается для кубита за состояние  $|0\rangle$ , а проекция (-1) – за состояние  $|1\rangle$ .

**Пример 2.** Берётся раствор молекул и помещается при комнатной температуре во внешнее магнитное поле. При этом атомные ядра, обладающие ядерным спином, т.е. являющиеся как бы маленькими магнитами, займут одно из двух положений – по полю – это  $|0\rangle$  и против него – это  $|1\rangle$ .

### Состояние квантового регистра в момент времени $t$

Разряд квантового регистра находится в состоянии

$$\alpha|0\rangle + \beta|1\rangle,$$

$$\alpha, \beta \in \mathbb{C}.$$

Поэтому состояние квантового  $m$ -разрядного регистра – это когерентная суперпозиция всех базисных состояний:

$$|\psi(t)\rangle \equiv \sum_{n_{m-1}=0}^1 \sum_{n_{m-2}=0}^1 \dots \sum_{n_0=0}^1 c_{n_{m-1}n_{m-2}\dots n_0} |n_{m-1}n_{m-2}\dots n_0\rangle, \tag{3}$$

$$c_{n_{m-1}n_{m-2}\dots n_0} \in \mathbb{C}.$$

Числа  $|c_{n_{m-1}n_{m-2}\dots n_0}|^2$  интерпретируются как вероятность пребывания регистра в состоянии  $|n_{m-1}n_{m-2}\dots n_0\rangle$ .

Например, состояние

$$|\psi(t)\rangle \equiv c_1 \underbrace{|01001011100011\dots\rangle}_m +$$

$$+ c_2 \underbrace{|11001011100011\dots\rangle}_m + \dots$$

**Комментарий.** Состояние 1-разрядного регистра квантового компьютера

в момент времени  $t$  подобно одновременному пребыванию кота в живом и мёртвом состоянии:

$$|\text{живо-мёртвое}\rangle = \alpha |\text{живо}\rangle + \beta |\text{мёртвое}\rangle$$

Рис. 1. Живо-мёртвое состояние кота

Иначе говоря, в квантовом мире альтернативы могут существовать одновременно.

Если предположить, что квантовый компьютер находится сразу во множестве параллельных вселенных, то в одной вселенной кот жив, а в другой мёртв. Наблюдатель видит только того кота, в какой вселенной живет сам; параллельный, другой мир он не видит. Такой подход называется эвереттовской интерпретацией квантовой механики [2].

### 3. Процессор

Процессор компьютера служит для того, чтобы менять состояние регистров.

Делается это посредством физического воздействия на биты в классическом компьютере и на кубиты – в квантовом; в результате и те и другие меняют свое состояние.

**Пример.** Если кубит представляет собой атом, то регистр – это квантовая система из  $m$  атомов. Воздействие на эту систему осуществляется с помощью специально подобранных импульсов лазеров. Лазерные импульсы влияют на электронные состояния атомов.

Лазерными импульсами управляет уже классический компьютер, входящий в состав того устройства, которое мы назвали квантовым компьютером.

#### 3.1. Классический процессор

Процессор классического компьютера состоит из схем, собранных из логических элементов.

*Логический элемент* – это простейшее устройство ЭВМ, выполняющее одну определенную логическую операцию над входными сигналами согласно правилам алгебры логики.

Для логических элементов независимо от их физической реализации приняты дискретные значения входных и выходных сигналов; обычно это два уровня, которые условно принимаются за 0 и 1.

На рис. 2 изображен логический элемент «НЕ», который переводит 0 в 1 и 1 в 0.

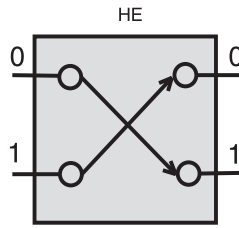


Рис. 2. Логический элемент «НЕ»

Логические элементы – это технические устройства, реализующие некоторые логические операции классической логики. Так, элемент «НЕ» соответствует операции  $\neg$ , элемент «ИЛИ» – операции  $\vee$  и т.д. Установлено, однако, что элементная база классического компьютера основывается всего на двух логических элементах, например на «НЕ» и «исключающее ИЛИ-НЕ».

Процессор преобразует, меняет содержание разрядов регистра посредством каждого входящего в него логического элемента  $U$ :

$$U : |n_{m-1}n_{m-2}\dots n_0\rangle \rightarrow |n'_{m-1}n'_{m-2}\dots n'_0\rangle. \quad (4)$$

### 3.2. Квантовый процессор

Квантовый процессор также состоит из логических элементов, называемых *гейтами*.

На рис. 3 изображен квантовый логический элемент « $\sqrt{\text{НЕ}}$ », который переводит 0 в 1 и 1 в 0, а также 0 в 0 и 1 в 1, но лишь с вероятностью  $p_{ij} = 1/2$ , ( $i, j = 0, 1$ ).

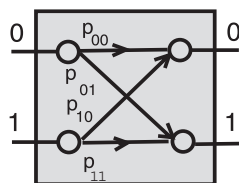


Рис. 3. Логический элемент с вероятностями  $p_{ij}$  переходов  $i \rightarrow j$ .

В теории квантового компьютера существует бесконечное количество логических элементов. Однако доказано, что квантовый компьютер может быть построен всего из двух логических элементов: однокубитового  $\widehat{Q}(\theta, \varphi)$  и 2-кубитового « $\widehat{CNOT}$ » (управляемое «НЕ»).

Квантовый процессор преобразует, меняет содержание разрядов квантового регистра посредством каждого входящего в него логического элемента (гейта)  $\widehat{U}$ :

$$\widehat{U} : \sum_{n_{m-1}=0}^1 \sum_{n_{m-2}=0}^1 \dots \sum_{n_0=0}^1 c_{n_{m-1}n_{m-2}\dots n_0} |n_{m-1}n_{m-2}\dots n_0\rangle \rightarrow$$

$$\rightarrow \sum_{n_{m-1}=0}^1 \sum_{n_{m-2}=0}^1 \dots \sum_{n_0=0}^1 c_{n_{m-1}n_{m-2}\dots n_0} |n'_{m-1}n'_{m-2}\dots n'_0\rangle. \quad (5)$$

Как видно из формулы (5), **в один шаг изменены сразу все  $2^m$  значений базисных состояний**. Это *эффект параллелизма* в работе квантового компьютера, не имеющий места для классических компьютеров. Для такой производительности за один шаг потребовалось бы  $2^m$  классических процессоров. Если  $m = 16$ , то  $2^{16} = 65536!$

Состояние (3) называется *сцепленным*, если оно не может быть представлено в виде

$$|x_1\rangle \dots |x_m\rangle, \quad (6)$$

где  $|x_j\rangle = \alpha_j|0\rangle + \beta_j|1\rangle$  ( $j = 1, \dots, m$ ).

Если состояние регистра (3) не является сцепленным, то это означает фактическое наличие в распоряжении для вычислений классического регистра вида (6), а значит, только он и преобразуется на данном такте работы квантового компьютера. Отсутствуют другие, параллельные базовые состояния, и, следовательно, отсутствует эффект квантового параллелизма, существенно ускоряющего работу компьютера и определяющего беспрецедентную эффективность квантовых вычислений.

#### 4. Условия для того, чтобы появился квантовый компьютер

Для того чтобы квантовый компьютер стал реальным инструментом для вычислений, необходимо решить следующие технические проблемы:

- Создать физическое устройство, содержащее достаточно большое число  $N > 100$  кубитов;
- Научиться приводить входной регистр в исходное основное базисное состояние

$$|\underbrace{00\dots 0}_m\rangle;$$

- Обеспечивать большое время декогеренции (не менее, чем в  $10^4$  раза больше времени выполнения основных квантовых операций (время такта)).

*Декогеренция* – это взаимодействие системы кубитов с окружающей средой. Она приводит к разрушению суперпозиций квантовых состояний и делает невозможным выполнение квантовых алгоритмов.

- Обеспечить возможность измерения состояния квантовой системы на выходе, то есть при выводе результата.



Рис. 4. Устройство квантового компьютера [1].

## 5. Вычисление на квантовом компьютере

### 5.1. Ввод начальных данных

Дано базовое состояние регистра (памяти):

$$|\underbrace{00\dots 0}_m\rangle \equiv \underbrace{|0\rangle \otimes |0\rangle \otimes \dots \otimes |0\rangle}_m. \quad (7)$$

С помощью последовательного применения к состоянию (7) гейтов

$$\hat{U}^{(1)}, \hat{U}^{(2)}, \dots, \hat{U}^{(m)},$$

$$\hat{U}^{(k)} = \hat{I} \otimes \dots \otimes \hat{I} \otimes \underbrace{\hat{U}_1}_k \otimes \hat{I} \otimes \dots \otimes \hat{I},$$

где  $\hat{U}^{(k)}$  действует только на  $k$ -й кубит посредством гейта  $\hat{U}_1$ , преобразующего однокубитовое состояние, квантовый регистр приводится в  $m$ -кубитовое состояние, являющееся *когерентной суперпозицией* всех базисных состояний:

$$\begin{aligned} \hat{U}^{(m)} \hat{U}^{(m-1)} \dots \hat{U}^{(1)} : |\underbrace{00\dots 0}_m\rangle &\rightarrow \hat{U}^{(m)} \hat{U}^{(m-1)} \dots \hat{U}^{(1)} |\underbrace{00\dots 0}_m\rangle = \\ &= \sum_{n=0}^{2^m-1} c_n |n\rangle \equiv \\ &\equiv \sum_{n_{m-1}=0}^1 \sum_{n_{m-2}=0}^1 \dots \sum_{n_0=0}^1 c_{n_{m-1}n_{m-2}\dots n_0} |n_{m-1}n_{m-2}\dots n_0\rangle, \end{aligned} \quad (8)$$

где

$$n = (n_{m-1}n_{m-2}\dots n_0)_2 = \sum_{l=1}^m n_{m-l} 2^{m-l}$$

– двоичное представление числа  $n$  и

$$\sum_{n=0}^{2^m-1} |c_n|^2 = 1.$$

Состояние (8) является начальным. Ввод информации завершен.

**Пример.** Если

$$\hat{U}_1|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \quad \hat{U}_1|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle),$$

то

$$\hat{U}^{(m)}\hat{U}^{(m-1)}\dots\hat{U}^{(1)}|\underbrace{00\dots 0}_m\rangle = \frac{1}{\sqrt{2^m}} \sum_{n=0}^{2^m-1} |n\rangle. \quad (9)$$

(Здесь все  $m$ -кубитовые базовые состояния  $|n\rangle$  равновероятны).

Состояние (9) является начальным. Ввод информации завершен.

## 5.2. Вычисление

Вычисление – это преобразование  $\hat{U}_F$  начального состояния (8):

$$\hat{U}_F \left( \sum_{n=0}^{2^m-1} c_n |n\rangle \right) = \sum_{n=0}^{2^m-1} c_n \hat{U}_F |n\rangle = \sum_{n=0}^{2^m-1} c_n |F(n)\rangle. \quad (10)$$

В случае (9) имеем

$$\hat{U}_F \left( \frac{1}{\sqrt{2^m}} \sum_{n=0}^{2^m-1} |n\rangle \right) = \frac{1}{\sqrt{2^m}} \sum_{n=0}^{2^m-1} |F(n)\rangle. \quad (11)$$

Конкретная реализация преобразования  $\hat{U}_F$  представляет собой запрограммированный квантовый алгоритм вычисления значений функции  $F$ .

Как видно из формулы (10), **в один шаг вычислены сразу все значения функции  $F$** . Это *эффект параллельности* квантовых вычислений, о котором мы говорили в § 2.2.

## 5.3. Вывод результата

Вывод результата в квантовом компьютеринге – это *измерение* квантового состояния (10):

$$\sum_{n=0}^{2^m-1} c_n |F(n)\rangle \rightarrow |F(n)\rangle.$$

В силу принципа квантовой механики вмешательство измеряющего устройства (устройство вывода данных) означает декогеренцию, т.е. разрушение когерентного состояния (10). Мы получаем значение  $F(n)$  лишь с вероятностью  $|c_n|^2$ .

В нашем примере (8) с равной вероятностью  $1/2^m$  любое значение  $F(n)$ !



Получаемый на выходе результат вычислений вследствие декогеренции, как видим, носит *вероятностный характер!* Иначе говоря, то, что получено на выходе, – состояние (регистра)  $|F(n)\rangle$  – верно лишь с некоторой вероятностью  $|c_n|^2$ .

«Наблюдение (части) памяти – не то же самое, что «печать результата». Мы должны спланировать серию прогонов одной и той же квантовой программы и последующую классическую обработку наблюдаемых результатов, и мы можем только надеяться получить желаемый результат с вероятностью, близкой к единице» [3, с.271].

## 6. Исправление квантовых ошибок

Классические компьютеры надежны, поскольку производимые вычисления можно защитить от *сбоев*, т.е. от ошибок, возникающих вследствие воздействия окружающей среды.

Взаимодействие квантового компьютера с окружающей средой ведет к декогеренции, которая разрушает когерентную суперпозицию и тем самым останавливает то, что делает квантовые вычисления привлекательными по сравнению с классическими, – их параллельность.

Возникновение декогеренции – то же, что появление сбоев в работе классических ЭВМ, поэтому борьбу с декогеренцией, её преодоление называют *исправлением квантовых ошибок*.

Декогеренцию, а также *квантовый шум*, т.е. взаимодействие  $m$ -кубита  $|q\rangle$  со средой  $\mathcal{E}$ , можно представить в виде:

$$|q\rangle|\mathcal{E}_0\rangle \rightarrow \sum_k \widehat{E}_{i_k}|q\rangle|\mathcal{E}_k\rangle, \quad (12)$$

где

$|\mathcal{E}_0\rangle$  – состояние среды до взаимодействия,  
 $\widehat{E}_j$  –  $j$ -й оператор ошибки (тип ошибки, один из трех, т.е.  $j = 1, 2, 3$ ),  
 $|\mathcal{E}_k\rangle$  –  $k$ -е состояние среды после взаимодействия.

*Исправление квантовых ошибок* – это процесс  $Cr$ , организованный в ходе работы квантового компьютера, который переводит состояния вида  $\widehat{E}_{i_k}|q\rangle$  в  $|q\rangle$ .

В результате имеем восстановление чистого состояния регистра  $|q\rangle$ :

$$\sum_k \widehat{E}_{i_k}|q\rangle|\mathcal{E}_k\rangle \xrightarrow{Cr} |q\rangle|\mathcal{E}_f\rangle,$$

свободного от помех (сцепленности со средой).

Разработаны различные методы исправления квантовых ошибок.

## 7. Классический компьютер вычисляет всё, что вычисляет квантовый

Изменения во времени состояния регистра  $|\psi(t)\rangle$  квантового компьютера описываются с помощью основного уравнения квантовой механики – уравнения Шрёдингера:

$$i\hbar \frac{d}{dt} |\psi(t)\rangle = \hat{H} |\psi(t)\rangle.$$

Здесь  $\hat{H}$  – гамильтониан, реализующий конкретный вычислительный (квантовый) алгоритм.

Как известно, решение уравнения Шрёдингера можно записать в виде

$$|\psi(t)\rangle = e^{-\frac{i}{\hbar} \int_0^t \hat{H} dt} |\psi(0)\rangle.$$

Это квантовая эволюция начального регистра  $|\psi(0)\rangle$ . Отсюда видно, что найти  $|\psi(t)\rangle$  можно, производя классические вычисления экспоненты от матрицы. Это крайне трудоёмкие вычисления, но, в принципе, выполнимые. Следовательно, классический компьютер может вычислить всё, что вычисляет квантовый компьютер, и нет никакого шанса построить квантовый компьютер, вычисляющий классически невычислимые функции.

## ЛИТЕРАТУРА

1. Валиев К.А., Кокин А.А. Квантовые компьютеры: надежды и реальность. М.: Ижевск: РХД, 2001.
2. Гуц А.К. Основы квантовой кибернетики: Учебное пособие. Омск: Полиграфический центр КАН, 2008. 204 с
3. Манин Ю.И. Классическое и квантовое вычисление и факторизация Шора / Квантовый компьютер и квантовые вычисления. Ижевск: НИЦ «Регулярная и хаотическая динамика», 2001.
4. Манин Ю.И. Вычислимое и невычислимое. М.: Советское радио, 1980.
5. Фейнман Р. Моделирование физики на компьютерах / Сб.: Квантовый компьютер и квантовые вычисления. Ред. ж-ла «Регулярная и хаотическая динамика». Ижевск, 1999. С.96-124.
6. Фейнман Р. Квантовомеханические компьютеры / Сб.: Квантовый компьютер и квантовые вычисления. Ред. ж-ла «Регулярная и хаотическая динамика». Ижевск, 1999. С.125-156.