**УДК 537.611**

**V.G. DEIBUK, G.P. GORSKYI**

# PRIMINIVE LOGICAL ELEMENTS SYNTHESIS BY QCE EMULATOR

***Yu.Fedkovych Chernivtsi National University,***
***2, Kotsiubynsky str., 58012, Chernivtsi, Ukraine,***
***E-mail: gena_grim@mail.ru***

**Анотація.** Метою статті є синтез і дослідження в середовищі емулятора QCE ряду найпростіших традиційних і нових квантових логічних елементів та алгоритмів.

**Аннотация.** Целью статьи является синтез и исследование в среде эмулятора QCE ряда простейших традиционных и новых квантовых логических элементов и алгоритмов.

**Abstract.** The aim of present paper is emulation and exploration of primitive quantum logical elements by QCE emulator tools. This emulator is based on Izing quantum computer hardware model.

**Keywords:** quantum computer, emulator, logical elements.

## PROBLEM ACTUALITY AND INVESTIGATION AIM

Today quantum computation problem is one of key problems of computer sciences [1]. It became actual problem after proving of $n$-tuple quantum register parallelism. Due to this phenomenon $n$-tuple quantum register during single machine instruction may contain and process not one but $2^n$ n-digit numbers simultaneously. Sense of parallelism is in fact, that quantum bit (QB) is not in fixed state logical 0 or 1 (if its state is not fixed especially), but uniformly in two states, thus average bit value is 0.5. Such quantum parallelism may be used for quick solving of many problems, such as search in large disordered databases (Grover's algorithm), or big integers factorization (Shore's algorithm). Actuality of last problem lies in sense of high-protected computer cryptosystems design. Quantum bits, which role may play different objects, such as electron or nuclear spin aggregates, quantum dots massifs, Josephson junctions in superconductors and other, may be guided by electric or magnetic fields, optical radiation (e.g. laser beam), etc. But now direct experimental researches with such objects are expensive and require special equipment. In other hand theoretical computation and physical modeling are essential of every new microprocessor generation design. The aim of present article is synthesis and approbation of some quantum logical elements and quantum algorithms.

## QCE PRINCIPLES AND BASIC OPERATIONS

Especially for quantum computers (QC) and quantum algorithms (QA) designing, study and investigation purposes some variants of quantum computer emulators (QCE) are developed. One of it is emulator QC on nuclear magnetic resonance (NMR) [1]. It is realized by C++ programming language. Essence of emulation process is solving of time-dependent Schrödinger equation (TDSE) by Suzuki product method. This method is approximate, but its results are enough exact for most of practical purposes. TDSE is solving for aggregate of spins, which are in strong static and radio-frequency (RF) weak magnetic fields and interact one another by exchange field, which causes phenomenon of ferromagnetism.

As model Hamiltonian we use Izing Hamiltonian, which may be represented as:

$$H(t) = -\sum_{i \neq j} \sum_{\alpha=x,y,z} J_{ij}^{\alpha}(t) S_i^{\alpha} S_j^{\alpha} - \sum_{i} \sum_{\alpha=x,y,z} H_j^{\alpha}(t) S_j^{\alpha}, \quad (1)$$

where $J_{ij}^{\alpha}$ – exchange coupling integral between spins with numbers $i$ and $j$, $H_j^{\alpha}(t)$ – total magnetic field acting on spin number $j$ along $\alpha$ – axis direction. $H_j^{\alpha}(t)$ dependence is:

$$H_j^{\alpha}(t) = H_{0j}^{\alpha} + H_{1j}^{\alpha} \cos\left(2\pi f_j^{\alpha} t + \varphi_j^{\alpha}\right), \quad (2)$$

where $H_{0j}^{\alpha}$ – static part of magnetic field, $H_{1j}^{\alpha}, f_j^{\alpha}, \varphi_j^{\alpha}$ – respectively amplitude, frequency and initial phase of RF magnetic field, which acts on spin number $j$ along $\alpha$ – axis direction.

All QA, which may be designed and checked by this QCE, are sets of microinstructions (MI) in which framework the external static and RF and internal exchange magnetic fields acting on spins (which represent QB) and duration of their acting are determined.

MIs are the "electronic blanks", in which corresponding cells we type values of Hamiltonian parameters according to desired action on spins. Action of external and internal magnetic fields on spins may be represented in terms of their rotations by user defined angles around fixed axes. Model Hamiltonian parameters are not enough for rotation angle definition. We must define the MI action time. For this purpose in "electronic blank" are the corresponding check boxes named "Main step", "Intermediate steps" and "Time step". Usually we take Main step=Intermediate steps=1. Thus MI action time is determined by Time step value.

Basic QC operation is spin (QB) number j rotation by arbitrary angle around fixed axis $\alpha$. Note, that this rotation may be realized if magnetic field is directed along rotation axis. If time is measured in full phase units, i.e. $2\pi$, we may take corresponding static magnetic field $H_{0j}^{\alpha}=1$ ($\alpha=x,y,z$). Thus executing time of MI $\tau=\varphi_0/2\pi$, where $\varphi_0$ is rotation angle in radians.

In order to illustrate interaction between QBs we describe operation $R_{12}(\varphi_0)$, which realizes controlled spin rotation by predetermined angle around $z$-axis. It acts in such manner. Guided spin (QB) 1 rotates around z-axis by $\varphi_0$ angle if guiding spin (QB) 2 is in logical 1 state and remains in initial state in opposite case. In order to include interaction between QBs at first we turn guiding QB by $\pi/2$ angle around y-axis clock wise. This is first MI $Y2$. Second MI $A_{12}(\varphi_0)$ realizes controlled rotation of QB 1.Its parameters are: $J_{12}^{x}=-1, H_{01}^{x}=0.5$, and action time is $\tau=\varphi_0/2\pi$. After that guiding QB must be returned to initial state. This may be realized by inverse transformation $\overline{Y}_2$. Thus we have $R_{12}(\varphi_0)=\overline{Y}_2 A_{12}(\varphi_0) Y_2$.

Another important QC operation is controlled phase shift. Its parameters are $J_{ij}^{z}=J, H_{0i}^{z}=H_{0j}^{z}=-J/2, \tau=-\varphi_0/J$. This operation doesn't influence on logical 0 or 1 probabilities for acting QB's. This property is important for quantum interference using in QA based on discrete Fourier transformation (DFT). Quantum interference has full analogy with interference of waves in optics.

Attractive peculiarities of this emulator are its physical clearness and free of charge circulation in Internet for education purposes.

Emulator has demonstration set of algorithms, but user may design and approbate different QA by using graphic user interface and option of generation approbation results as numerical data file for further mathematic processing.

## MAIN RESULTS AND ITS DISCUSSION

Now we describe some synthesized traditional and new primitive elements. They are: CNOT gate, swapping gate, 3-QB Toffoli gate, which may be used as AND gate, coincidence circuit and XOR gate on 3-QB rotation operation basic, OR gate, trigger and "smart" swapping gate. Yet we synthesized and approbated Grover's database search algorithm for 8 records database and $a^x \bmod N$ period "indicator".

Consider, e.g. CNOT gate. It's one of basic quantum gates. If QB 1 is controlling and QB 2 is target, that CNOT operation may be represented as $CNOT_{12}=\overline{Y}_2 I_{12} Y_2$, where $Y_2, \overline{Y}_2$ are QB rotations by angles $\pm\pi/2$ around $y$ axis, $I_{12}$ - operation with parameters $J_{12}^{z}=-1, H_{01}^{z}=0.5, \tau=0.5$. Its physical sense is QB 2 by angle $-\pi$ rotation around z-axis. Magnetic field compensates exchange field if controlling QB is in logical 0 state and amplifies it if controlling QB is in logical 1 state. Thus QB 2 inverts if QB 1 is in logical 1 state and doesn't invert in opposite case.

By CNOT gates combination we can design ordinary SWAP gate as $SWAP_{12}=CNOT_{12}CNOT_{21}CNOT_{12}$.

We may design the inverse CNOT gate. It may be represented as $CNOT_{12}^{(inv)}=\overline{Y}_2 \overline{I}_{12} Y_2$, where $\overline{I}_{12}$ is operation with parameters $J_{12}^{z}=1, H_{01}^{z}=-0.5, \tau=0.5$. This inverse gate inverts target QB if control QB is in logical 0 state and not inverts target QB in opposite case.

Toffoli gate repeats first and second input QB and inverts third input QB if first and second inputs QB both are in logical 1 state. In other cases Toffoli gate repeats third input QB too. If input QBs are 1,2,3 and output ones are 4,5,6 in reduced form Toffoli gate may be represented as

$$LOGTOF_{123-456} = R_6^{(y)}(4) I_{26} R_6^{(y)}(-4) I_{16} R_6^{(y)}(4) I_{26} R_6^{(y)}(4) I_{36} \overline{Y}_6 Y_5 I_{25} \overline{Y}_5 Y_4 I_{14} \overline{Y}_4, \quad (3)$$

where $R_6^y(4)$ and $R_6^y(-4)$ are rotations of QB 6 by angles $\pi/4$ or $-\pi/4$ respectively.

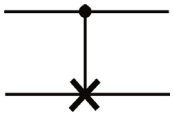

Fig. 1. CNOT-gate

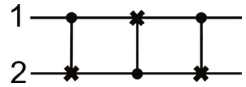


Fig.2. Ordinary swapping gate

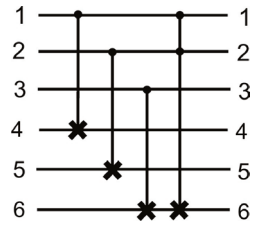


Fig.3. Toffoli gate for 6 QBs

If initial state of output QB 3 is logical 0 then XOR-NOT gate (coincidence circuit) may be represented as:

$$(XOR - NOT)_{123} = \overline{Y}_3 I_{123} Y_3, \quad (4)$$

where operation $I_{123}$ has parameters $J_{13}^z = J_{23}^z = 1, \tau = 0.5$. It means, that ordinary XOR gate may be represented as:

$$XOR_{123} = Y_3 I_{123} Y_3 = \overline{Y}_3 I_{123} \overline{Y}_3. \quad (5)$$

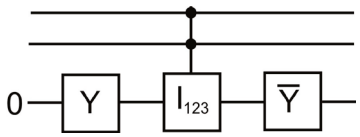


Fig.4. Coincidence circuit

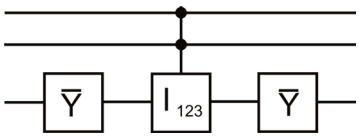


Fig. 5. XOR gate

By CNOT and Toffoli gate using we may synthesize elements OR, AND as well as trigger. Let QBs 1, 2 are input and QB 3 is output. Install QB 3 into logical 0.In this case AND gate may be represented as:

$$AND = TOF_{1,2-3}. \quad (6)$$

Now install QB 3 into logical 1.In this case OR gate may be represented as:

$$OR = Y_1^2 Y_2^2 TOF_{(1,2)-3} \overline{Y}_1^2 \overline{Y}_2^2. \quad (7)$$

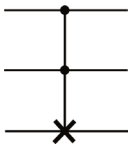

Fig. 6. AND or simple Toffoli gate

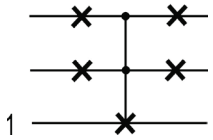


Fig. 7. OR gate

Now we describe the trigger. Let QB 1 is information input (D), QB 2 is enable input (E), QB 3 is direct output (Q) and QB 4 is inverse output ( $\overline{Q}$ ). If enable input is in logical 0 state trigger repeats input information (QB1) on direct output and inverts it on inverse output. In opposite case outputs 3 and 4 are swapping. This may be represented as:

$$TRG_{(1,2)-(3,4)} = Y_4 I_{24} \overline{Y}_4 Y_3 I_{23} \overline{Y}_3 \overline{Y}_4 I_{14} \overline{Y}_4 Y_3 I_{13} \overline{Y}_3. \quad (8)$$

Corresponding between input and output states of trigger may be represented by table 1.

Let us consider yet "smart swapping gate", which changes values of two QBs, if they are different and not changes it in opposite case. It is really XOR-circuit with feedback. QB 1 and 2 are input.

If they are different, that XOR(1,2)=1 and XOR(1,2)=0 in opposite case. If QB3 (auxiliary), which is output for XOR we use as controlling QB for QBs 1 and 2, we obtain such resulting representation for "smart swapping gate":

$$SmSWAP = CNOT_{31} CNOT_{32} XOR_{12} = \overline{Y}_1 I_{13} Y_1 \overline{Y}_2 I_{23} Y_2 Y_3 I_{123} Y_3. \quad (9)$$

Table 1.

Corresponding between input and output states of trigger

| QB1 (D) | QB2 (E) | QB3 (Q) | QB4 ( $\overline{Q}$ ) |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 |

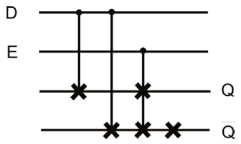

Fig. 8. Trigger

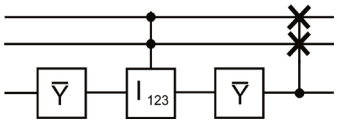

Fig. 9. "Smart" swapping gate

Now we consider Grover's algorithm [2] for 3-QB database, which contains 8 3-QB words (records). Its basic operation is controlled phase shift by $\pi/8$ angle. Parameters of corresponding MI $g_{12}$ are $J_{12}^{z}=1$, $\tau=0.25$. QA consist of 3 stages: preparing 3-tuple register superposition state, labeling of records and search of records. This may be represented as:

$$GROV_j=\overline{Y}_3X_3\overline{Y}_2X_2\overline{Y}_1X_1g_{12}^3f_jg_{12}^3\overline{Y}_1\overline{X}_1^2\overline{Y}_2\overline{X}_2^2\overline{Y}_3\overline{X}_3^2. \quad (10)$$

Here $f_j$ are the labeling functions for records with numbers 0, 1…7. They may be represented as it's shown in table 2

Table 2.

Labeling functions for different records

| Number of record, $j$ | Labeling function $f_j$ representation |
|---|---|
| 0 | $\overline{Y}_3X_3\overline{Y}_2X_2\overline{Y}_1X_1$ |
| 1 | $\overline{Y}_3X_3Y_2\overline{X}_2\overline{Y}_1\overline{X}_1$ |
| 2 | $\overline{Y}_3X_3Y_2X_2\overline{Y}_1X_1$ |
| 3 | $\overline{Y}_3X_3\overline{Y}_2\overline{X}_2\overline{Y}_1\overline{X}_1$ |
| 4 | $\overline{Y}_3\overline{X}_3\overline{Y}_2X_2\overline{Y}_1X_1$ |
| 5 | $\overline{Y}_3\overline{X}_3Y_2\overline{X}_2\overline{Y}_1\overline{X}_1$ |
| 6 | $\overline{Y}_3\overline{X}_3Y_2X_2\overline{Y}_1X_1$ |
| 7 | $\overline{Y}_3\overline{X}_3\overline{Y}_2\overline{X}_2\overline{Y}_1\overline{X}_1$ |

Now consider alternative version of Grover's algorithm for 3 QBs. It has the form:

$$GROV_j=\overline{Y}_3X_3\overline{Y}_2X_2\overline{Y}_1X_1g_{12}g_{13}g_{23}\tilde{f}_jg_{12}g_{13}g_{23}\overline{Y}_1\overline{X}_1^2\overline{Y}_2\overline{X}_2^2\overline{Y}_3\overline{X}_3^2. \quad (11)$$

Labeling functions for it are represented in table 3.

Table 3.

Labeling functions for records in alternative algorithm

| Number of record, $j$ | Labeling function representation $\tilde{f}_j$ |
|---|---|
| 0 | $\overline{Y}_3\overline{X}_3\overline{Y}_2\overline{X}_2\overline{Y}_1\overline{X}_1$ |
| 1 | $Y_3\overline{X}_3Y_2\overline{X}_2\overline{Y}_1\overline{X}_1$ |
| 2 | $\overline{Y}_3X_3Y_2X_2\overline{Y}_1X_1$ |
| 3 | $\overline{Y}_3X_3\overline{Y}_2\overline{X}_2\overline{Y}_1\overline{X}_1$ |
| 4 | $Y_3X_3\overline{Y}_2X_2\overline{Y}_1X_1$ |
| 5 | $\overline{Y}_3\overline{X}_3Y_2\overline{X}_2\overline{Y}_1\overline{X}_1$ |
| 6 | $Y_3X_3Y_2X_2\overline{Y}_1X_1$ |
| 7 | $\overline{Y}_3\overline{X}_3\overline{Y}_2\overline{X}_2\overline{Y}_1\overline{X}_1$ |

But last algorithm is not exact. Simulation shows, that algorithm represented by (10) generates numbers of records $\{0,1,2,3,4,5,6,7\}$ in binary notation. Algorithm (11) generates "average" numbers {0.875, 1.625, 2.375, 3.125, 3.875, 4.625, 5.375, 6.125}, or numbers $\{0,1,2,3,4,5,6,7\}$ with probability 0.67. It's quantum effect, which hasn't classical analog.

Consider yet "quantum emulator" of classical Shore's algorithm. It's not quantum Shore's algorithm in the true sense. But it's "indicator" of period $a^x \bmod N$ function. Since $a^x \bmod N = 1$, if $x = 0$, then period of this function is first positive integer, for which $a^x \bmod N = 1$. Let $N = 15$, $a = 2$. Hence we have $x = 4$. Now we obtain $15 = (2^2 + 1)(2^2 - 1) = 5 \cdot 3$. In binary notation $4 = 100$, $1 = 001$. Let use QBs 4, 5, 6 for indication of $a^x \bmod N$ period and QBs 1,2,3 for indication $a^x \bmod N$ value if $a^x \bmod N = 1$. Thus for the case $N = 15$, $a = 2$ indicator performance algorithm may be represented as:

$$f = \overline{Y}_1 I_{41} Y_1 \overline{Y}_2 I_{52} Y_2 \overline{Y}_3 I_{63} Y_3^3 Y_2^2 . \qquad (12)$$

Values of cubits for this case are represented in table 4.

Table 4.

Values of cubits for different $x$ if $f(x) = 2^x \bmod 15$

| $x$ | QB1 | QB2 | QB3 | QB4 | QB5 | QB6 |
|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| 2 | 1 | 1 | 1 | 0 | 1 | 0 |
| 3 | 0 | 1 | 1 | 1 | 1 | 0 |
| 4 | 1 | 0 | 0 | 0 | 0 | 1 |

From this table we see, that period of $f(x) = 2^x \bmod 15$ is equal to 4 because only for this case we obtain 1 (in binary notation) in "indicating" QBs 1, 2, 3.

## CONCLUSIONS

1. In Izing quantum computer hardware model framework we emulated and investigated some primitive quantum logical elements. Standard ones are CNOT and TOFFOLI GATE. Last one may be considered as universal logical element. New ones are "traditional" classical computer logical elements, as coincidence circuit XOR-NOT, XOR, AND, OR, trigger and introduced by authors "smart" swapping gate.
2. For XOR-NOT, XOR and "smart" swapping gate synthesis we introduced the new 3-QB rotation operation $I_{123}$.
3. Yet we emulated and explored some algorithms, as two Grover's algorithm versions for 8 records database and period indication algorithm for $a^x \bmod N$ function in case $a = 2, N = 15$.

## REFERENCES

1. Валиев К.А. Квантовые компьютеры и квантовые вычисления / К.А.Валиев // Успехи физических наук. – 2005. – Т.175, №1. – С.5 – 39. *Библиогр.: с.38-39.*
2. Mishielsen K., De Raedt H. QCE: A Simulator For Quantum Computer Hardware / K. Mishielsen, H. De Raedt // Turk. J. Phys. – 2003. – V.27. – P.1 – 29. *Ref.:p.28-29.*
3. Крупичка С. Физика ферритов и родственных им магнитных окислов: в 2 т. / С. Крупичка*;* [*пер. с нем. под ред. А.С. Пахомова*]. – М: Мир, 1976. – *Перевод по изд. Svatopluk Krupička. Physic der ferrite und verwandten magnetischen oxide (Prag: Academia, 1973)*,Т.1. – 1976. – 354 с**.** *Библиогр. в конце глав.*
4. Бауместер К. Физика квантовой информации. – М: Постмаркет, 2002.

Надійшла до редакції 23.11.2008р.

**DEIBUK V. G. — Sc.D., Professor of Computer Systems and Networks Department, Chernivtsi National University, Chernivtsi , Ukraine, vdei@chnu.edu.ua.**

**GORSKYI G. P. — master's degree holder, Chernivtsi National University, Chernivtsi National University, Chernivtsi , Ukraine, gena_grim@mail.ru.**