ISSN 2278-3091

Volume 9 No.2, March - April 2020

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse130922020.pdf

https://doi.org/10.30534/ijatcse/2020/130922020

# Simulation of Quantum Computation via MAGMA Computational Algebra System

**Binh A. Nguyen[1], Viet Q. Tran[1], Khoa D. Ta[1], Manh Hoang[1],Thien V. Truong[1], Nhan D. Nguyen[2], Duc M. Nguyen[3]**

[1]ICT Department, FPT University, Hanoi, Vietnam, binhnase04865@fpt.edu.vn, viettqse06178@fpt.edu.vn, khoatdhe130813@fpt.edu.vn, manhhhe130294@fpt.edu.vn, thientvse04522@fpt.edu.vn

[2]Dept. of Biomedical Engineering, Sungkyunkwan University, Suwon, South Korea, nhannd@skku.edu

[3]School of Electrical Engineering, University of Ulsan, Ulsan, Korea, nguyenmanhduc18@gmail.com

**ABSTRACT**

Quantum computation is the usage of quantum mechanics to process information. It has been proven that quantum computation has the capacity to transfer data securely based on its fundamental and it solves the complex issues better than classical computation. However, before the quantum computers are available for people living, we need to verify and build up some frameworks for verification of quantum information system. In this research, we consider the **MAGMA** system which can be useful for quantum computation and calculation parameters of quantum stabilizer codes. Therein, we consider the problem of the first quantum error correction codes and we verify that basic algorithm on **MAGMA** system. The proposed system prove that **MAGMA** can be used for many tasks of quantum algorithms, quantum communication, and quantum computation problems.

**Key words:** Quantum computation, **MAGMA** tool, Shor code.

## 1. INTRODUCTION

Quantum information system is a system which is based on quantum mechanical phenomena, such as superposition and entanglement to perform operates on a system state. The computation algorithms based on quantum computer have proved the efficient on processing data more security and solving complex problem more efficient time [1, 18-22]. For example, one of the first quantum algorithm to factor an integers into its primers is invented by Shor[2], which runs on polynomial time. Moreover, Grover [3] proposed a searching algorithm named Grover search, which is applied on many reality research on large database system. Hence, quantum computers are attracted by many researchers all over the world [4, 5, 6].

However, the efforts tobuild them have been hampered by the fragility of qubits since they areeasily affected by heat and electromagnetic radiation. This type of error is called decoherence and the field of error correction in quantum computation examines the different ways to avoid decoherence. Since the first discussion of quantum code(QECC) was invented by Shor [7], the theory of QECC is generalization to be expressed as the quantum stabilizer code. Therefore, the importance of QECC on practical building of quantum computer is no longer in doubt [8, 9, 10, 11].

Before implementing the algorithms, computation on quantum computer, the necessary step is to simulate them on the classical computer. Among many quantum systems models such as quantum circuits model, quantum adiabatic computation, Zidan's model [12,13,14,15], which are proven to have those effective on simulation of quantum algorithm, quantum protocol, quantum communication. Quantum circuit model is related to mathematical model and it is suitable choice for verification of quantum computation. Hence, in this research, we study a framework which is based on **MAGMA** algebra computation environment to simulate and analysis two basic solution of error correction code on quantum information system.

We organize this study as follows. Section 2 will review basic problem of quantum computers such as quantum bits, operations of quantum bits. In section 3, **MAGMA** system is introduced, and the implementation of quantum algorithms based on **MAGMA** is explained. Finally, the conclusion is listed in Section 4.

## 2. QUANTUM INFORMATION AND QUANTUM COMPUTATION

Quantum systems use qubit which simulates two levels system such as: atoms, ions, electrons or protons and their respective control devicesthat are working together to act as computer memory. A qubit $|\varphi\rangle = \varphi_1|0\rangle + \varphi_2|1\rangle$ is considered to be found in the both basis states $|0\rangle$ and $|1\rangle$ where the probability value of state $|0\rangle$ is $|a|^2$ and of state $|1\rangle$ is $|b|^2$. It is called the superposition concept of a qubit, which is one of a main property of quantum information since the amount of information which presented in qubits are no limitation. The matrix form of a qubit can be represented in Hilbert space as,

$$|\varphi\rangle = \begin{bmatrix}\varphi_1\\ \varphi_2\end{bmatrix} = \varphi_1\begin{bmatrix}1\\0\end{bmatrix} + \varphi_2\begin{bmatrix}0\\1\end{bmatrix} = \varphi_1|0\rangle + \varphi_2|1\rangle. \quad (1)$$

According to norm condition for a qubit on the Bloch sphere space, the complex numbers $a$ and $b$ satisfy the equation $|\varphi_1|^2 + |\varphi_2|^2 = 1$. A $n$ qubits system is constructed by multiple tensor products of some other qubits, it is given as follows,

$$|\phi\rangle = \sum_{i=0}^{2^n-1} \phi_i |i\rangle = \sum_{i_k=\{0,1\}} \phi_{i_1 i_2 \dots i_n} |i_1\rangle|i_2\rangle \dots |i_n\rangle. \quad (2)$$

where $i = \sum_{j=0}^{n-1} 2^j i_j$.

| Gate | Notation | Matrix |
|---|---|---|
| NOT ( Pauli-$X$) | $X$ | $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ |
| Pauli-$Z$ | $Z$ | $\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ |
| Hadamard | $H$ | $\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ |
| CNOT ( Controlled NOT ) | | $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ |

**Figure 1:**Basic Gates in quantum computation.

Note that the condition for any quantum gate is revertible and the invert gate that move $\mathbf{U}|\psi\rangle$ back to $|\psi\rangle$ satisfy $\mathbf{U}^{-1} = \mathbf{U}^{\dagger}$, so **U** is unitary matrix. **Fig**. 1 shows the basis of quantum gates, all the quantum gates can be represented as their linear combination. Pauli channel of quantum system consists of four basic elements, namely **X**, **Z**, **Y**, and**I** (identity matrix). Any operation and errors acting on qubit can be represented as the combination of them. Hence, we have three types of errors: bit flip, phase flip, and their combination. In general, the error operators that effect on $n$ qubits have the form: $E = e_1 \otimes e_2 \otimes \dots \otimes e_n$ where $e_i \in \{\mathbf{I}, \mathbf{X}, \mathbf{Y}, \mathbf{Z}\}$.

There are many quantum computation models can be used to explain the quantum state and quantum computation. Among them, quantum circuit model is related to mathematical model and it is suitable for simulation in **MAGMA**environment.

## 3. SIMULATION OF QUANTUM COMPUTATION OVER MAGMA SYSTEM

### *A. MAGMA system*

**MAGMA** is a software tool which we can install in PC or we can used as web-based for the purpose of computation in number theory, algebra, algebraic geometry, and combinatorics. It is open access for study, research purpose and provides a comfortable defined environment for many problems of mathematical such as graph, group, fields, code designs, and many others [16]. Using **MAGMA**, we can do the working with quantum computation since it offers some basic tool for defining the quantum state, Hilbert space, Galois field, and unitary transformation of quantum states. In this study, we use the web-based **MAGMA** tool, it is free and can be found at [17].

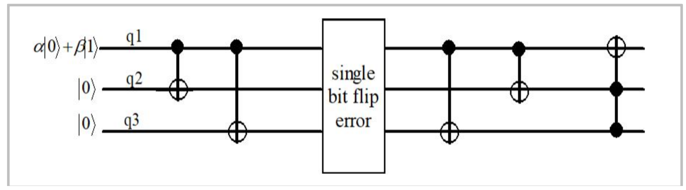


**Figure 2:**Quantum bit flip error correction code.

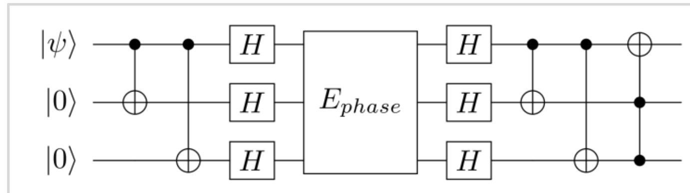


**Figure 3:**Quantum phase flip error correction code.

### *B. Simulation of Quantum Repetition Code*

The simplest and first QECC is three qubits repetition code for bit flip or phase flip error. **Fig.** 2, 3 are the quantum circuit models for correction those types of errors. The only difference between two circuits is the using of Hadamard matrix on correction of phase flip error, which is since the property of Clifford gates: **Z**=**HXH** and **X**=**HZH**.

The **MAGMA** programs of quantum circuits for **Fig.** 2, 3 are described as follows. First, the information qubits are declared. The quantum system starts with the initial information, we extend it to the 3-qubits system via helps of ancilla 2 qubits of zeros, after transformation by encode step, the logical states or encoded qubits are created. Then, the quantum gates as previous mentioned must to be declared. Here, the quantum gates **Bit-flip**, **Phase-flip**, and **Controlled-NOT** gates are used. Here, the encoding state is as follows,

$$|\mathbf{0}_L\rangle = |000\rangle, |\mathbf{1}_L\rangle = |111\rangle. \quad (4)$$

### *C. Simulation of Shor code*

To extend the first full quantum code, Shor code for 9 qubits is created by Shor, which use both bit-flip correction and phase-flip correction and can correct bit-flip, phase-flip, and their combination. To do so, for one qubit is protected against phase-flip we need extend it to codeword of three qubits. Then, each qubits of that three-qubits need to extend to three-qubits to protect against bit-flip error. Hence, the quantum circuit starts with the initial information, we extend it to the 9-qubits via helps of ancilla 8 qubits of zeros, after transformation by encode step, the logical states or encoded qubits are created. Here, two basis states of codewords 9-qubits repetition as follows,

$$|\mathbf{0}_L\rangle = \frac{1}{\sqrt{8}}(|000\rangle + |111\rangle)\otimes(|000\rangle + |111\rangle)\otimes(|000\rangle + |111\rangle),$$

$$|\mathbf{1}_L\rangle = \frac{1}{\sqrt{8}}(|000\rangle - |111\rangle)\otimes(|000\rangle - |111\rangle)\otimes(|000\rangle - |111\rangle).$$

Using matrices transformation, the states after applying error and decoding can be found. The final states show us the correction state can be recovered the syndrome $|S_e\rangle$tell us which error has applied to logical states. The full quantum circuit for Shor code is given in **Fig.** 5. And the **MAGMA**program for Shor code is described in **Fig.** 6.

```
% Repetition code: For bit flip error:
F<i> := ComplexField(4);
H1 := HilbertSpace(F, 3);
f := 3/5 * H1![0,0,0] + 4/5 * H1![1,0,0];
f;
ControlledNot(~f, {1}, 2);
ControlledNot(~f, {1}, 3);
f;
BitFlip(~f, 2);
f;
ControlledNot(~f, {1}, 2);
ControlledNot(~f, {1}, 3);
ControlledNot(~f, {2,3}, 1);
f;
%Result
0.6000|000> + 0.8000|100>
0.6000|000> + 0.8000|111>
0.6000|010> + 0.8000|101>
0.6000|010> + 0.8000|110>
```

```
% Repetition code: For PhaseFlip error:
F<i> := ComplexField(4);
H1 := HilbertSpace(F, 3);
f := 3/5 * H1![0,0,0] + 4/5 * H1![1,0,0];
f;
ControlledNot(~f, {1}, 2);
ControlledNot(~f, {1}, 3);
f;

f1 := BitFlip(f, 1);
f2 := PhaseFlip(f, 1);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;
f1 := BitFlip(f, 2);
f2 := PhaseFlip(f, 2);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;
f1 := BitFlip(f, 3);
f2 := PhaseFlip(f, 3);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;

PhaseFlip(~f, 3);

f1 := BitFlip(f, 1);
f2 := PhaseFlip(f, 1);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;
f1 := BitFlip(f, 2);
f2 := PhaseFlip(f, 2);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;
f1 := BitFlip(f, 3);
f2 := PhaseFlip(f, 3);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;

ControlledNot(~f, {1}, 2);
ControlledNot(~f, {1}, 3);
ControlledNot(~f, {2,3}, 1);
f;
%Result
0.6000|000> + 0.8000|100>
0.6000|000> + 0.8000|111>
0.5999|001> + 0.8000|101>
```

**Figure 4:**Repetition codeon MAGMA system.

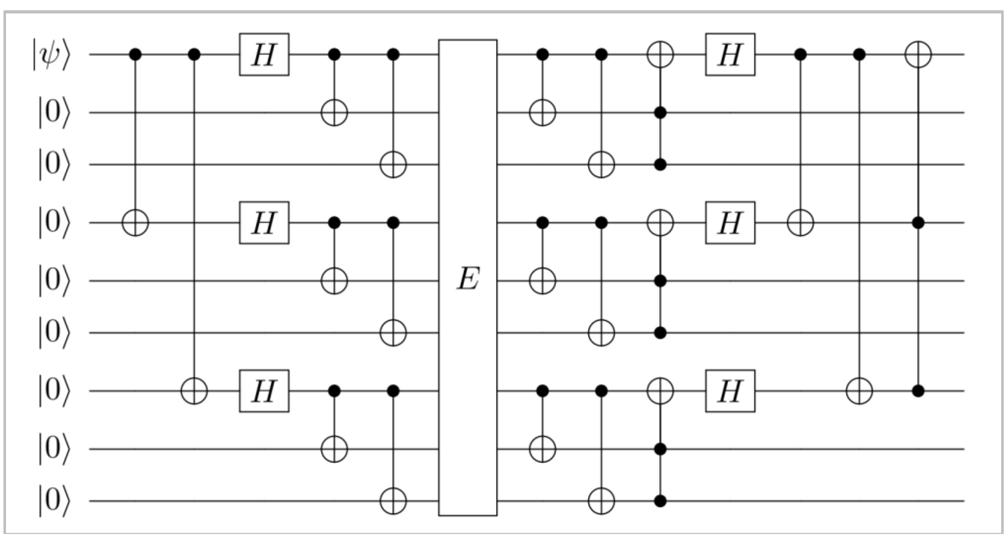


**Figure 5:**Quantum full bit flip, phase flip error correction.

```
% Shor code:
F<i> := ComplexField(4);
H1 := HilbertSpace(F, 9);
f := 3/5 * H1![0,0,0,0,0,0,0,0,0] + 4/5 * H1![1,0,0,0,0,0,0,0,0];
f;

ControlledNot(~f, {1}, 4);
ControlledNot(~f, {1}, 7);
f;

f1 := BitFlip(f, 1);
f2 := PhaseFlip(f, 1);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;
f1 := BitFlip(f, 4);
f2 := PhaseFlip(f, 4);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;
f1 := BitFlip(f, 7);
f2 := PhaseFlip(f, 7);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;
f;

ControlledNot(~f, {1}, 2);
ControlledNot(~f, {1}, 3);
ControlledNot(~f, {4}, 5);
ControlledNot(~f, {4}, 6);
ControlledNot(~f, {7}, 8);
ControlledNot(~f, {7}, 8);
f;

PhaseFlip(~f, 3);
BitFlip(~f, 3);
f;

ControlledNot(~f, {1}, 2);
ControlledNot(~f, {1}, 3);
ControlledNot(~f, {4}, 5);
ControlledNot(~f, {4}, 6);
```

```
ControlledNot(~f, {7}, 8);
ControlledNot(~f, {7}, 8);
ControlledNot(~f, {2,3}, 1);
ControlledNot(~f, {5,6}, 4);
ControlledNot(~f, {8,9}, 7);
f;

f1 := BitFlip(f, 1);
f2 := PhaseFlip(f, 1);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;
f1 := BitFlip(f, 4);
f2 := PhaseFlip(f, 4);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;
f1 := BitFlip(f, 7);
f2 := PhaseFlip(f, 7);
f := 1/SquareRoot(2)*f1 + 1/SquareRoot(2)*f2;
f;

ControlledNot(~f, {1}, 4);
ControlledNot(~f, {1}, 7);
ControlledNot(~f, {4,7}, 1);
f;
%Result
0.6000|000000000> + 0.8000|100000000>
0.6000|000000000> + 0.8000|100100100>
0.4949|000000000> - 0.07074|100000000> - 0.07074|000100000> +
0.4949|100100000>
- 0.07074|000000100> + 0.4949|100000100> + 0.4949|000100100> -
0.07074|100100100>
0.4949|000000000> - 0.07074|111000000> - 0.07074|000111000> +
0.4949|111111000>
- 0.07074|000000100> + 0.4949|111000100> + 0.4949|000111100> -
0.07074|111111100>
0.07074|110000000> + 0.4949|001000000> - 0.4949|110111000> -
0.07074|001111000>
- 0.4949|110000100> - 0.07074|001000100> + 0.07074|110111100> +
0.4949|001111100>
0.4949|001000000> + 0.07074|101000000> - 0.07074|001100000> -
0.4949|101100000>
- 0.07074|001000100> - 0.4949|101000100> + 0.4949|001100100> +
0.07074|101100100>
0.5999|101000000> + 0.8000|001100100>
0.5999|001100100> + 0.8000|101100100>
```

**Figure 6:** Shor code on MAGMA system.

## 4. CONCLUSION

The paper presents basic information on quantum information system these are qubits, unitary transformation. In addition, we use **MAGMA** computational algebra system with web-based tool for a verification of the three-qubits repetition and nine-qubits Shor code. Such verification of simplest QECC help us better understanding of quantum error correction and quantum algorithm.

The outstanding result prove that the proposed framework is novel for further researches simulation of quantum information system. In the future, we plan to use this framework for simulation of quantum stabilizer codes, quantum algorithms, and quantum communication.

## ACKNOWLEDGEMENT

This work is supported by FPT University, Hanoi, Vietnam; Sungkyunkwan University, Suwon, Republic of Korea; and University of Ulsan, Ulsan 44610, Republic of Korea(by the Research Program through the National Research Foundation of Korea NRF-2019R1A2C1005920).

**Conflict of Interest:** On behalf of all authors, the corresponding author declares that there is no conflict of interest.

## REFERENCES

1. Bennett, C.H., Shor, P.W. **Quantum information theory**. (1998). IEEE Transaction on Information theory. 44(6):2724-2742. https://doi.org/10.1109/18.720553
2. Shor, P.W. **Algorithms for quantum computation: discrete logarithms and factoring.** (1994). Proceedings 35th Annual Symposium on Foundations of Computer Science. IEEE Comput. Soc. Press.
3. Grover, L.K. **Quantum Mechanics Helps in Searching for a Needle in a Haystack**. (1997). Phys. Rev. Lett. 79:325. https://doi.org/10.1103/PhysRevLett.79.325
4. Nguyen, D.M., Kim, S. **Quantum Key Distribution Protocol Based on Modified Generalization of**

**Deutsch-Jozsa Algorithm in d-level Quantum System**. (2019). Int. J. Theor. Phys. 58(1):71-82. https://doi.org/10.1007/s10773-018-3910-4

5. Nguyen, D.M., Kim, S. **Multi-Bits Transfer Based on the Quantum Three-Stage Protocol with Quantum Error Correction Codes.** (2019). Int. J. Theor. Phys. 58(6):2043-2053. https://doi.org/10.1007/s10773-019-04098-4
6. Nguyen, D.M., Kim, S. **The fog on Generalized teleportation by means of discrete-time quantum walks on N-lines and N-cycles.** (2019). Modern Physics Letters B. 33(23):1950270. https://doi.org/10.1142/S0217984919502701
7. Peter W. Shor, **Scheme for reducing decoherence in quantum memory.** (1995). Phys. Rev. A. 52(4).
8. Nguyen, D.M., Kim, S. **Quantum stabilizer codes construction from Hermitian self-orthogonal codes over GF(4).** (2018). Journal of Communications and Networks. 20(3):309-315. https://doi.org/10.1109/JCN.2018.000043
9. Nguyen, D.M., Kim, S. **New Constructions of Quantum Stabilizer Codes Based on Difference Sets**. (2017). Symmetry. 10(11):655.
10. Nguyen, D.M., Kim, S. **New construction of binary and nonbinary quantum stabilizer codes based on symmetric matrices.** (2019). International Journal of Modern Physics B. 33(24):1950274. https://doi.org/10.1142/S0217979219502746
11. Nguyen, D.M., Kim, S. **Construction and complement circuit of a quantum stabilizer code with length 7.** (2016). Eighth International Conference on Ubiquitous and Future Networks (ICUFN). 332 - 336.
12. Noson.S. Yanofsky, Micro.A. Mannucci, **Quantum computing for computer scientists**. (2008). Cambridge University Press New York, NY, USA. https://doi.org/10.1017/CBO9780511813887
13. Zidan, M., et. al. **A Novel Algorithm based on Entanglement Measurement for Improving Speed of Quantum Algorithms.** (2018). Appl. Math. Inf. Sci. 12(1):265-269. https://doi.org/10.18576/amis/120127
14. Zidan, M., et. al. **Quantum Classification Algorithm Based on Competitive Learning Neural Network and Entanglement Measure.** (2019). Appl. Sci. 9(7):1277.
15. Zidan, M., et. al. **A quantum algorithm based on entanglement measure for classifying Boolean multivariate function into novel hidden classes**. (2019). Results in Phys. 15:102549. https://doi.org/10.1016/j.rinp.2019.102549
16. http://magma.maths.usyd.edu.au/magma/handbook/text/1942#21811
17. http://magma.maths.usyd.edu.au/calc/
18. K. Ravi, N. K. Pandy, A. S. Kumar, and N. Kushal. **Quantum Computer-Hardware**, International Journal of Advanced Trends in Computer Science and Engineering, Vol. 5, No. 4, pp. 46-53, 2016.
19. K. Ravi, N. K. Pandy, A. S. Kumar, and N. Kushal. **Quantum Computer-Algorithms**, International Journal of Advanced Trends in Computer Science and Engineering, Vol. 5, No. 4, pp. 54-60, 2016.
20. A. Naincy, B. Pratap. **Develop a Hybrid Method to Encode Data**, International Journal of Advanced Trends in Computer Science and Engineering, Vol. 6, No. 3, pp.51-56, 2017.
21. Nguyen, D.M., Kim, S. **A novel construction for quantum stabilizer codes based on binary formalism**. (2020). International Journal of Modern Physics B. 34(8): 2050059. https://doi.org/10.1142/S0217979220500599
22. Nguyen, D.M., Kim, S. **Quantum stabilizer codes based on a new construction of self-orthogonal trace-inner product codes over GF(4)**. (2020). International Journal of Modern Physics B. 34(5):2050017. https://doi.org/10.1142/S0217979220500174