Design and Synthesis of Efficient Circuits for Quantum Computers

Archimedes D. Pavlidis*

Department of Informatics & Telecommunications,
University of Athens, Athens, Greece,
{erevos@di.uoa.gr}

Abstract. The recent advances in the field of experimental construction of quantum computers with increased fidelity components shows that large-scale machines based on the principles of quantum physics are likely to be realized in the near future. As the size of the future quantum computers will be increased, efficient quantum circuits and design methods will gradually gain practical interest. The contribution of this thesis towards the design of efficient quantum circuits is two-fold. The first is the design of novel efficient quantum arithmetic circuits based on the Quantum Fourier Transform (QFT), like multiplier-with-constant-and-accumulator (MAC) and divider by constant, both of linear depth (or speed) with respect with the bits number of the integer operands. These circuits are effectively combined so as they can perform modular multiplication by constant in linear depth and space and consequently modular exponentiation in quadratic time and linear space. Modular exponentiation and modular multiplication operations are integral parts of the important quantum factorization algorithm of Shor and other quantum algorithms of the same family, known as Quantum Phase Estimation algorithms. Important implementation problems like the required high accuracy of the employed rotation quantum gates and the local communications between the gates are effectively addressed. The second contribution of this thesis is a generic hierarchical synthesis methodology for arbitrary complex and large quantum and reversible circuits. The methodology can handle more easily larger circuits relative to the flat synthesis methods. The proposed method offers advantages over the standard hierarchical synthesis which uses Bennett's method of "compute-copy-uncompute".

Keywords: Quantum computer architectures, quantum arithmetic circuits, quantum Fourier transform, quantum circuits synthesis, reversible circuits synthesis

1 Introduction

Quantum Information Theory and Quantum Computing are interdisciplinary research fields that combine different doses of Physics, Informatics and Mathematics depending on which aspect someone focuses. Quantum Computing is

* Dissertation Advisor: D. Gizopoulos, Professor

a relatively recent research field, although Quantum Information Theory has already been developed for the last 40 years, after important results which connect classical Information Theory to Quantum Mechanics (quantum entropies inequalities [2, 18, 19], Holevo bounds for capacities of quantum channels [15, 16], Bekenstein bound [5], etc.)

The theoretical connection of Quantum Mechanics to the Theory of Computation achieved in the 80's [11, 12], while more boost came in the 90's with the invention of efficient quantum algorithms [32, 30, 14], which can be executed on computing machines (quantum computers) exploiting fundamental quantum properties of nature, like superposition and entanglement. Such efficient algorithms can achieve important reduction of time complexity, so that in many instances, problems that cannot be solved in polynomial time on a classical computer with the currently known algorithms, can be solved in polynomial time on a quantum computer. A famous example, with important applications in Cryptography, is the factorization of a composite integer into its prime factors (Shor's algorithm)[30]. Another important example is the efficient simulation of quantum physical systems with many degrees of freedom (like a complex chemical molecule), a computation which is not practically achievable in a classical computer [20].

The physical realization of a quantum computer, while in principle is feasible, requires a complex technological effort to overcome practical problems. An important problem is that the carriers of quantum information, the qubits, are very fragile under the influence of their environment and it is very difficult to maintain them in a constant state for a long enough duration so as they can perform a useful computation. The physical carriers of information can be atoms, ions, nuclei and in general any microscopic system on which quantum mechanical effects can be observed[1]. The disturbance effect on the qubits under the environment influence is known as decoherence and can be thought as an environment noise effect. Decoherence problems increase as the number of qubits increases. Additionally, the basic processing elements of qubits, the quantum gates, introduce another factor of disturbance of quantum information, because usually their operation approximates the ideal theoretical operation with errors which don't allow the construction of useful large quantum computers. These introduced errors can be thought as an additional environment induced noise, converting the ideal gates to noisy or erroneous ones. Thus, although real quantum computers have been already developed using various technologies (photons, ion traps, Josephson junctions), they are limited to about 10 qubits [21, 3, 27, 35].

The decoherence problem has been theoretically addressed in the 90's by exploiting and extending results from classical Error Correcting Codes Theory, leading to the invention of Quantum Error Correcting Codes [31, 7, 33]. Such codes can be applied by combining many noisy quantum physical gates so as to build an ideal quantum logical gate, that is they allow the construction of

[1] Currently, some of the most promising are ion traps [9] and Josepshon junction superconductors [36]

fault tolerant quantum gates. This can be accomplished under some conditions, of which the most important is that the noise percentage introduced by each physical quantum gate is lower than a threshold (Quantum Threshold Theorem) [1]. In such a case, an ideal quantum logical gate can be constructed by using redundancy, that is using many physical gates. During the recent years, the effort to build high reliability quantum gates has been intensified, so as to permit the construction of quantum computers of adequate size in the near future. Results of these efforts are very encouraging.

This thesis contributes two-fold:

1. Design of novel efficient quantum circuits (arrays of interconnected quantum logical gates) for integer arithmetic operations and their combination to a higher hierarchy level to achieve more complex arithmetic operations, like modular exponentiation which is an integral part of Shor's algorithm and important algorithms of the same class [24] . The novelty of the proposed circuits lays in the usage of Quantum Fourier Transform (QFT) on the integers states prior to their processing, resulting in improved efficiency in terms of speed. Problems related to the usage of QFT in arithmetic circuits, such as the requirement for high precision quantum gates and the lack of communications locality between the qubits, are also effectively addressed.
2. A generic hierarchical quantum and reversible circuits synthesis methodology [25, 26]. The majority of existing automatic synthesis methods are flat; they operate on the lowest level of gates and while in many cases they lead to optimal or suboptimal results, they have the disadvantage of not being suitable for large circuits as they have exponential requirements in memory usage and run time. The straightforward incorporation of hierarchical synthesis methods into tools of flat methods uses the methodology of Bennett. In contrast, the proposed hierarchical method offers advantages in terms of derived circuit speed and memory, relative to the few hierarchical ones of the literature.

2 Design of Novel Efficient Quantum Circuits

In the context of this thesis, the used gates are assumed to be reliable (logical level) which have been derived from elementary physical quantum gates incorporating any method of error correction. Thus, the thesis concerns the logical level of quantum gates and not the lower level of physical gates. Therefore, the proposed methods of this doctoral thesis can be applied to any technology of physical realization and fault tolerant implementation of logic gates.

We adopt the computation speed, which is known as circuit depth, as the main criterion of efficiency of the proposed methods in this thesis, and it is the number of required steps to complete the computation. This is an important efficiency criterion when construction of large size, in terms of memory, quantum computers become feasible in the future.

The proposed quantum subsystems concern basic arithmetic operations on integers, like multiplication of a constant with an integer and accumulation

(ΦMAC) and division by constant (GMΦDIV) (quotient and remainder calculation) which are used in important quantum algorithms. The implementations is accomplished by using alternative representation of integers in the Fourier domain (that is we use the Quantum Fourier Transform) instead of the usual representation in the computational basis. Quantum circuits using QFT exist in the literature, but they are limited to various kind of adders only [13], while the straightforward implementation of a MAC with Fourier representation using such adders [4] has quadratic circuit depth relative to the integer size. In contrast, the proposed ΦMAC offers linear depth, a considerably important property for large (and thus practically useful) quantum numbers. Regarding the division circuits, just a few quantum dividers exist in the literature and they are chiefly limited to special purposes (e.g. for Galois fields $GF(2^m)$, that is dividers of polynomials with coefficients 0 and 1). A known general quantum divider based on QFT [17] has a cubic depth, while if the divisor is constant its depth can be reduced to be quadratic. The proposed constant divider in this thesis offers a linear depth.

The above two circuits, effectively combined, can be used to construct other more complex circuits useful in various important quantum algorithms. In this thesis we show how it is possible to construct a constant multiplier modulo N (ΦMULMOD), which is a fundamental element for the operation of modular exponentiation. Modular exponentiation is the most time consuming operation in one of the most important quantum algorithms, the factorization algorithm of Shor, and also in other algorithms of the same family. The proposed design achieves a circuit depth of $O(n^2)$, while the majority of the circuits in the literature ranges between $O(n^2 \log n)$ and $O(n^3)$, and consequently the proposed design offers important speed advantage for large numbers. Some of the circuits in the literature offering quadratic or less depth have the disadvantage of increasing excessively the required space (number of qubits) in order or they have the disadvantage of performing approximate calculation.

In the estimation of the circuit efficiency (being in time or space) we must take into account the physical implementation constraints. Such a constraint is the capability of global interactions between the qubits or the limitation of this interaction to neighborhood qubits only, e.g. in a linear one-dimensional array implementation of qubits, where each one can interact only with its two neighbors (1D-LNN, 1D-Linear Nearest Neighborhood). The proposed architecture for Shor's algorithm, while at first sight seems to require global communications between the qubits, it can be adapted in physical machines requiring local interactions with constant overhead in depth, as we show. That is, we don't have any increase in the quadratic order of depth. In contrast, most of the low $O(n^2 \log n)$ depth architectures when applied in a machine that requires local communications increase the depth (e.g. to $O(n^2\sqrt{n})$ in 2D-LNN or to $O(n^3)$ in 1D-LNN) [8].

The Fourier domain processing of the proposed circuits requires the usage of controlled rotation quantum gates with specific angles. A known drawback of such gates is that they do not belong to the category of gates that may constructed fault tolerantly, unless they are decomposed in a sequence of fault tolerant capable gates (e.g. H and T gates). But, such a decomposition implies considerable overhead in the depth of the whole modular exponentiation circuit up to an order, that is to $O(n^3)$ from $O(n^2)$. Yet, it is possible, as we show, to have a much lesser overhead of $O(n^2 \log n)$ by permitting approximate computation which allow the Shor's algorithm to operate with minor degradation concerning the probability of success. Therefore, the proposed architecture is one of the most competitive in terms of depth, especially if it is applied to 1D-LNN or 2D-LNN physical machines, which are the most probable to be implemented in the future.

3 Hierarchical Synthesis of Quantum and Reversible Circuits

Design of quantum circuits adopts ideas from classical logical design. Small circuits or circuits with repetitive structure can be designed either ad hoc or with formal synthesis methods based on specifications (e.g. truth tables). In the case of quantum circuits there exist similar synthesis methods based on specifications which in the general case are unitary matrices [10, 29]. In special cases where a quantum circuit is described by a matrix with elements exclusively 0 and 1, then reversible circuits[2] synthesis methods can be exploited [28]. Such quantum circuits cases are met when the circuit computes an arithmetic or logical function in the computational basis (e.g. integer addition).

In such cases, these methodologies are suitable for small circuits only, because the required computation power and memory required for their application increases exponentially with the circuit size. The obvious solution is the hierarchical bottom-up design which is applied in classical circuits. In the hierarchical method, if the desired operation can be described as a splicing of simpler operations, the design starts from the lowest level of simpler operations towards the higher level of the more complex operations. The application of the hierarchical method to quantum circuits is possible but requires special handling of the intermediate computation results that are not useful at the end. The particularity is caused due to the fact that these intermediate results cannot be simply discarded at the end because, in general, they are quantum entangled with the desired results. They must be reset to their initial state by inverse computation. Bennett's method is a well known method that keeps the desired results through copying and resets the intermediate results through uncomputation [6]. Its main characteristic and drawback is that it doubles the computation steps (forward computation and the reverse computation) and it also requires more memory space, equal to the space needed by the desired results due to the copying.

The proposed hierarchical synthesis method transforms the initial specifications of the quantum circuit which are given as arrays and arrays of list representing the classical sequence of operation into a directed acyclic graph called

[2] In a reversible circuit, for every possible output, the respective input can be derived, that is no information erasure happens [34].

forward Quantum Dependence Graph (QDG). The nodes of the forward QDG correspond to the components of a quantum library and they suppose to implement the elementary arithmetic operations. These components could be known constructions from the literature (adders etc), synthesized by other low level synthesis method, or populated by the proposed method applied to a lower level. The arcs connecting the QDG nodes correspond to qubits or quantum registers and they are discriminated in arcs which are affected by their successor node and the ones that control their successor node. The final qubits state of the derived forward QDG describes the desired result along garbage results produced during the computation.

The method adopted to reset the garbage states is to apply uncomputation locally on each node that really needs such an inversion of computation, instead to apply it globally as Bennett's method suggests. Namely, nodes of the forward QDG that are effectively involved in garbage production are marked (these are the nodes which have paths with affected arcs towards final garbage states). These marked nodes of forward QDG are traversed backwards and an inverse of each node is appended to the QDG. The inverse nodes are part of the library as it contains quantum circuits whose inverses are assured to exist.

However, data dependencies between the nodes may not always allow such an inversion, in which case we have a deadlock. Two special procedures are applied to detect and resolve such deadlocks (type I and II deadlocks) before the uncomputation stage. Both procedures have the cost to introduce additional ancilla qubits but they never exceed the additional ancilla qubits that would be needed if Bennett's method would be applied.

The proposed synthesis method requires polynomial execution time and memory space in relation to the number of the functions of the specifications and in any case it produces circuits of equal or better performance in terms of depth and space in compared to the basic Bennett's method.

4 Conclusions

Quantum arithmetic circuits based on the QFT representation of integers, instead of the usual computational basis representation, is an alternative implementation that may offer various advantages if used properly. This is due to the fact that two of the main core blocks are the constant adder, which has a constant depth of 1 when the computation is carried out in a datapath that contains an already QFT transformed integer, and the controlled constant adder which has a linear depth of n. By keeping a sequence of computations in such a datapath without reverting back to the computational basis it is possible to maintain a linear depth which otherwise would be impossible. This can be achieved by exploiting properties of the controlled rotation gates such as commutativity, decomposition and suitable rearrangement so as to pipeline their execution. The initial direct QFT and the final inverse QFT does not alter the linear depth as both transforms can be performed in linear depth. Thus, a computation level of hierarchy can be climbed onto (e.g. in our case, addition to multiplication), without any respective time complexity increase.

Another advantage of using QFT based arithmetic is the lower space requirements. This is manifested in Beauregard's modular exponentiation [4] , where $2n + 1$ qubits are adequate for the full Shor's algorithm. The reason is that no carry computations are needed in the QFT adder as this is done implicitly with the angle additions. While this advantage is not observed in the proposed modular exponentiation circuit due to the divider complexity, it remains in the multiplier/accumulator ΦMAC where no ancilla qubit is used. Also, robustness of such circuits to gate pruning and rotation angle approximation is observed in various instances.

All these remarks suggest that arithmetic circuits, like the proposed ones, are estimable as building blocks for larger and more complex arithmetic circuits.

The obvious follow up to the QFT arithmetic circuits would be to exploit them to derive more complex arithmetic circuits, useful for various quantum algorithms, like the constant divider was for Shor's algorithm. In the same branch of interest, the subject of approximate computations can be further investigated through simulations. The bounds reported in the thesis may be loose and better results may be obtained with numerical simulations. Numerical simulations of the full Shor's algorithm, like the ones performed in [22, 23], are difficult for the case of the proposed circuit because of the requirement of $8n + 2$ qubits. For example, to factor $N = 15$ we would need to simulate $8 \cdot 4 + 2 = 34$ qubits. The joint state vector of 34 qubits consists of $2^{34} \approx 16 \cdot 10^9$ complex elements leading to about 128Gbytes of memory when using single precision floating point, only for the state vector. Yet, partial simulations can be proven useful. A simulation to derive distances between the ΦMAC and an approximated ΦMAC are feasible ($3n+1$ qubits), or even a similar simulation for the whole divider ($6n+1$ qubits).

The hierarchical design method we propose in the doctoral thesis offers advantages relative to Bennett's method in terms of speed and memory of the target circuit. The specifications of the synthesizable circuit are given as a sequence of arithmetic or logic functions. These functions are supposed to be part of a library of quantum circuits. The library can be constructed by using other lower level synthesis methods, or contain known parametrized circuits of the literature (e.g. adders) or be populated with new circuits of the same hierarchical method. Also, the library contains the inverse circuits due to the necessity described above. The end result of the synthesis in the form of directed acyclic graph (Quantum Dependence Graph - QDG) describes the target circuit, where the nodes of the graph represent the modules of the library and the arcs of the graph represent the interconnections between the modules.

Regarding the hierarchical synthesis method, a next obvious step is to develop a complete software which would include front-end and back-end submodules. The front-end must be a compiler accepting the description of the classical algorithm in a suitable language and transforming it in the internal representation required by the synthesis algorithm. The back-end must combine the final QDG representation with information stored in the library so as to export the synthesized circuit in a low gate-level description such as in a quantum assembly format. Equipped with such an integrated tool, we could do a more systematic comparison with other high level synthesis tools, although the advantages of the proposed synthesis methodology are clear even without the tool.

References

1. Aharonov, D., Ben-Or, M.: Fault-tolerant Quantum Computation with Constant Error. In: Proc. 29th Annual ACM Symposium on Theory of Computing (STOC'97). pp. 176–188 (May 1997)
2. Araki, H., Lieb, E.H.: Entropy inequalities. Communications in Mathematical Physics 18(2), 160–170 (1970)
3. Barends, R., Lamata, L., Kelly, J., García-Álvarez, L., Fowler, A.G., Megrant, A., Jeffrey, E., White, T.C., Sank, D., Mutus, J.Y., Campbell, B., Chen, Y., Chen, Z., Chiaro, B., Dunsworth, A., Hoi, I.C., Neill, C., O'Malley, P.J.J., Quintana, C., Roushan, P., Vainsencher, A., Wenner, J., Solano, E., Martinis, J.M.: Digital quantum simulation of fermionic models with a superconducting circuit. Nature Communications 6, 7654:1–7654:7 (Jul 2015)
4. Beauregard, S.: Circuit for Shor's Algorithm Using $2n+3$ Qubits. Quantum Information & Computation 3(2), 175–185 (Mar 2003)
5. Bekenstein, J.D.: Universal upper bound on the entropy-to-energy ratio for bounded systems. Physical Review D 23, 287–298 (Jan 1981)
6. Bennett, C.H.: Logical Reversibility of Computation. IBM J. Research and Development 17(6), 525–532 (Nov 1973)
7. Calderbank, A.R., Shor, P.W.: Good quantum error-correcting codes exist. Physical Review A 54, 1098–1105 (Aug 1996)
8. Choi, B.S., Van Meter, R.: On the Effect of Quantum Interaction Distance on Quantum Addition Circuits. ACM J. Emerging Technologies in Computing Systems 7(3), 11:1–11:17 (Aug 2011)
9. Cirac, J.I., Zoller, P.: Quantum Computations with Cold Trapped Ions. Physical Review Letters 74, 4091–4094 (May 1995)
10. Cybenko, G.: Reducing Quantum Computations to Elementary Unitary Operations. J. Computing in Science and Engineering 3(2), 27–32 (Mar 1996)
11. Deutsch, D.: Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer. Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 400(1818), 97–117 (Jul 1985)
12. Deutsch, D.: Quantum Computational Networks. Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 425(1868), 73–90 (Sep 1989)
13. Draper, T.G.: Addition on a Quantum Computer. eprint arXiv:quant-ph/0008033 (Aug 2000)
14. Grover, L.K.: A Fast Quantum Mechanical Algorithm for Database Search. In: Proc. 28th Annual ACM Symposium on Theory of Computing (STOC'96). pp. 212–219 (May 1996)
15. Kholevo, A.S.: Bounds for the quantity of information transmitted by a quantum communication channel. Problemy Peredachi Informatsii (Problems of Information Transmission) 9(3), 3–11 (1973)
16. Kholevo, A.: On the capacity of a quantum communication channel. Problemy Peredachi Informatsii (Problems of Information Transmission) 15(4), 3–11 (1979)

17. Khosropour, A., Aghababa, H., Forouzandeh, B.: Quantum Division Circuit Based on Restoring Division Algorithm. In: Proc. 8th International Conference on Information Technology: New Generations (ITNG '11). pp. 1037–1040 (2011)
18. Lieb, E.H., Ruskai, M.B.: Proof of the strong subadditivity of quantum-mechanical entropy. Journal of Mathematical Physics 14(12), 1938–1941 (1973)
19. Lindblad, G.: Completely positive maps and entropy inequalities. Communications in Mathematical Physics 40(2), 147–151 (1975)
20. Lloyd, S.: Universal Quantum Simulators. Science 273(5278), 1073–1078 (Aug 1996)
21. Monz, T., Nigg, D., Martinez, E.A., Brandl, M.F., Schindler, P., Rines, R., Wang, S.X., Chuang, I.L., Blatt, R.: Realization of a scalable Shor algorithm. Science 351(6277), 1068–1070 (Mar 2016)
22. Nam, Y.S., Blümel, R.: Robustness and performance scaling of a quantum computer with respect to a class of static defects. Physical Review A 88, 062310 (Dec 2013)
23. Nam, Y.S., Blümel, R.: Streamlining Shor's algorithm for potential hardware savings. Physical Review A 87, 060304 (Jun 2013)
24. Pavlidis, A., Gizopoulos, D.: Fast Quantum Modular Exponentiation Architecture for Shor's Factoring Algorithm. Quantum Information & Computation 14(7&8), 649–682 (May 2014)
25. Pavlidis, A., Gizopoulos, D.: Hierarchical synthesis of quantum and reversible architectures. In: Proc. 12th ACM International Conference on Computing Frontiers (CF'15). pp. 13:1–13:8 (2015)
26. Pavlidis, A., Gizopoulos, D.: Hierarchical synthesis of quantum and reversible architectures. International Journal of Parallel Programming 44(5), 1028–1053 (Oct 2016)
27. Politi, A., Matthews, J.C.F., O'Brien, J.L.: Shor's quantum factoring algorithm on a photonic chip. Science 325(5945), 1221–1221 (Sep 2009)
28. Saeedi, M., Markov, I.L.: Synthesis and Optimization of Reversible Circuits - A Survey. ACM Computing Surveys 45(2), 21:1–21:34 (Feb 2013)
29. Shende, V.V., Bullock, S.S., Markov, I.L.: Synthesis of quantum-logic circuits. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 25(6), 1000–1010 (Jun 2006)
30. Shor, P.W.: Algorithms for Quantum Computation: Discrete Logarithms and Factoring. In: Proc. 35th Annual IEEE Symposium on Foundations of Computer Science, (FOCS'94). pp. 124–134 (Nov 1994)
31. Shor, P.W.: Scheme for reducing decoherence in quantum computer memory. Physical Review A 52, R2493–R2496 (Oct 1995)
32. Simon, D.: On the power of quantum computation. In: Proc. 35th Annual IEEE Symposium on Foundations of Computer Science, (FOCS'94). pp. 116–123 (Nov 1994)
33. Steane, A.: Multiple-Particle Interference and Quantum Error Correction. Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 452(1954), 2551–2577 (Nov 1996)
34. Toffoli, T.: Reversible computing. Tech. Rep. MIT/LCS/TM-151, Massachusetts Institute of Technology, Laboratory for Computer Science (Feb 1980)
35. Vandersypen, L.M.K., Steffen, M., Breyta, G., Yannoni, C.S., Sherwood, M.H., Chuang, I.L.: Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance. Nature 414(6866), 883–887 (Dec 2001)

36. Vion, D., Aassime, A., Cottet, A., Joyez, P., Pothier, H., Urbina, C., Esteve, D., Devoret, M.H.: Manipulating the Quantum State of an Electrical Circuit. Science 296(5569), 886–889 (May 2002)