Quantum Functional Devises and Quantum Computing

Hiroyuki Araki, Tadashi Ae
Graduate School of Engineering, Hiroshima University;
1-4-1 Kagamiyama, Higashi-Hiroshima, 739-8527 Japan;
Fax: +81-824-22-7195,
Email: ae@aial.hiroshima-u.ac.jp.

Abstract

We believe the quantum functional device to be a future perspective device, if we solve the problems that it has nowadays. We will summarize such problems with several discussions from the viewpoint of circuit and system.

Keywords: quantum functional device, quantum computer, quantum state, quantum memory

1 Introduction

The quantum functional device is expected as a new 21st century device, though there are many difficulties to realize it as a realistic functional device. In this paper we will discuss the circuit and system for the quantum functional device. When focusing on a quantum functional device, we encounter several constraints to realize it and doubt to realize the same circuit as one of conventional CMOS circuits. We must investigate the circuit/system realization from the start point and the wide viewpoint.

We believe the quantum functional device to be a future perspective device, if we solve the problems that it has nowadays. We will summarize such problems with several discussions.

2 System Constraint and Quantum Constraint

2.1 System Constraint

For a system consisting of conventional CMOS circuits, we usually encounter a constraint and summarize it as a system constraint for the general von-Neumann-type processor. The von-Neumann-type processor consists of operational circuits (usually, Arithmetic/Logic Units, i.e., ALUs), registers and control circuits.

Among processor designers, we call operational circuits and registers the data path, and control circuits the control path, respectively. For a system realization the control path is an overhead and we call it the system constraint. The system constraint includes

International Journal of Computing Anticipatory Systems, Volume 11, 2002
Edited by D. M. Dubois, CHAOS, Liège, Belgium, ISSN 1373-5411 ISBN 2-9600262-5-X

also the interconnections to realize the circuits. We need to pay an attention when designing a von-Neumann-type processor as an LSI chip, since the system constraint is usually an inevitable factor. The factor is large for the CISC (Complex Instruction Set Computer), and is relatively small for the RISC (Reduced Instruction Set Computer). The system constraint can be simply regarded as the interconnections and the switching circuits, and therefore, it becomes small for a special purpose chip. Even for such a chip design, however, we must keep the constraint in mind.

2.2 Quantum Constraint

We introduce

$$C \times \ell^2 = h \;\; (h: \text{a constant}),$$

where C is the minimum energy to move the state and ℓ is the design rule. We will have Figure 1 for a size-fixed chip, and also

$$N \times \ell^2 = const.,$$

where N is the system size (which is proportional to data size, i.e., memory bits). This is shown as in Figure 2.

As a result, we have

$$CN \times (\ell^2)^2 = const.,$$

as in Figure 3, where CN represents the minimum total energy for control for a size-fixed chip. It is noted that both x-axis and y-axis in Figures 1, 2 and 3 are drawn in log-scale.

Intuitively speaking, we encounter a large constraint when realizing a circuit using quantum functional devices (in short, q.f.d.'s), since the quantum device includes essentially a quantum phenomena. We introduce an assumption as follows;

$$CS > h \;\; (h: \text{a constant}),$$

where S is the state (i.e., potential energy), and C is the energy to move the state (i.e., moving energy). This assumption is similar to Heisenberg's Uncertainty Principle. We call it *quantum constraint*, and show it as in Figure 4 together with the system constraint, where x-axis and y-axis are the chip size (ℓ : design rule) and the memory size (bits which are proportional to C), respectively. Usually, the quantum constraint does not appear behind the system constraint, but it appears when the width on the chip design rule is less than 0.01μm, because the quantum constraint becomes larger than the system constraint.

From the viewpoint of circuit and system, we must consider a new system design for q.f.d.'s under the quantum constraint as well as the system constraint. This means that we will not desire a similar circuit/system to the conventional CMOS circuit/system, as a quantum functional circuit/system.

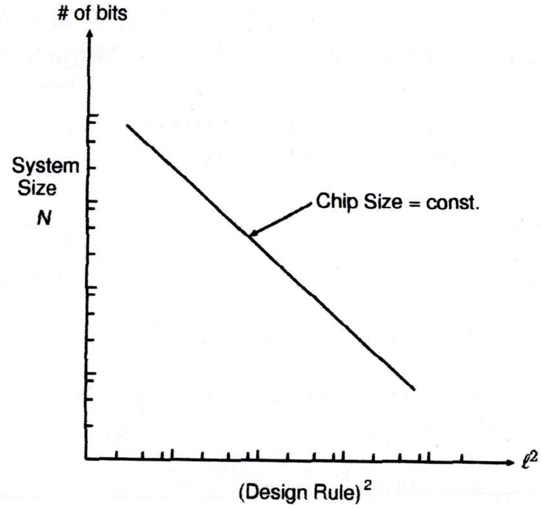


Fig. 1. System Size vs Design Rule when Chip Size is fixed.

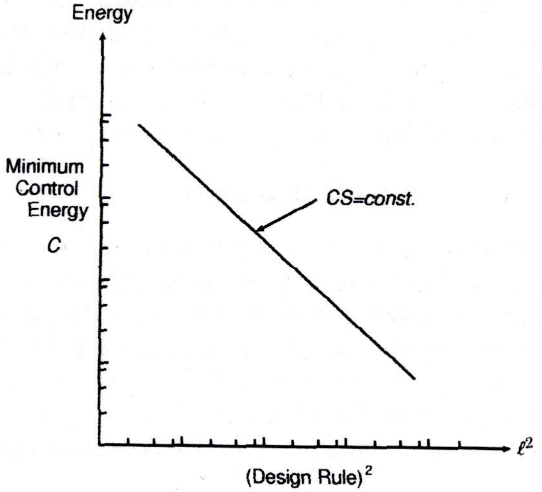


Fig. 2. Control Energy vs Design Rule.

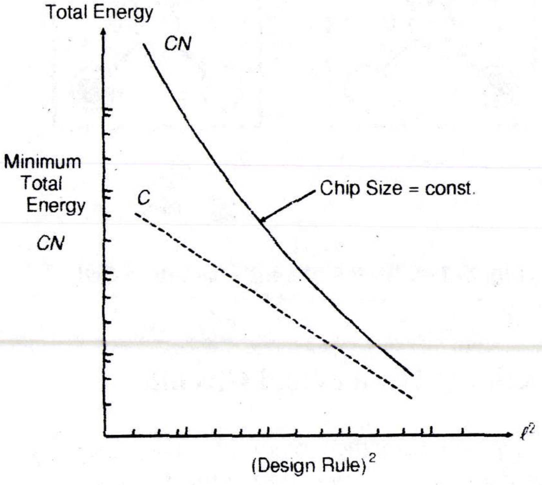


Fig. 3. CN vs Design Rule when Chip Size is fixed.

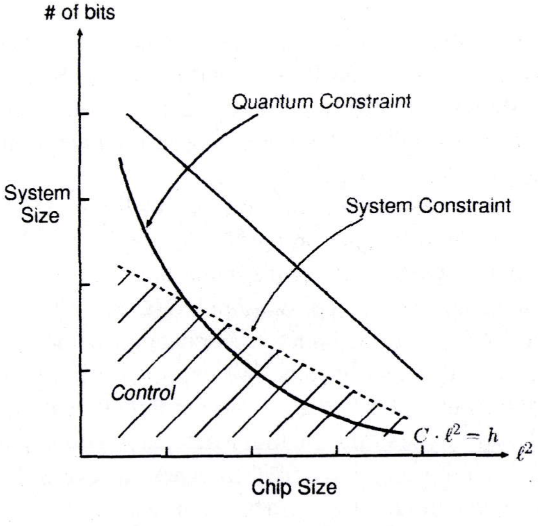
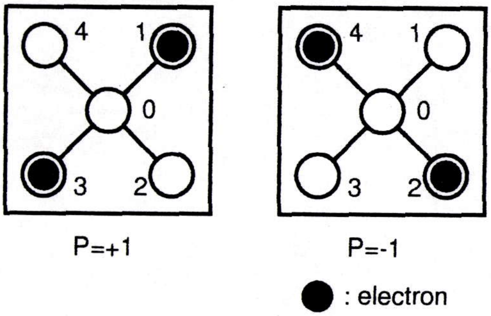


Fig. 4. System Constraint and Quantum Constraint.



P=+1 P=-1

Fig. 5. Two States of Lent's Quantum Cell.

3 Quantum Functional Devices and Circuits

Any q.f.d. will be acceptable from the circuit/system designer if it is really realizable. More precisely, the quantum dot, SET(Single Electron Transistor), etc. are all welcome, if the quantum state is "stable" for system application.

Let us focus on a quantum dot as an example. If we can realize the quantum dot as a memory, we need to make efforts to realize only the logic circuit. Lent's idea will be one of perspective realization of logic parts, where a two-stable latch circuit is realized without using any explicit interconnection. The idea of interconnection is regarded as a preferable one, since the realization of logic reduces also the overhead of interconnection.

A cell of five quantum dots, i.e., a quantum cell includes two states as in Figure 5 by two types of polarization (P=+1 and P=−1), and therefore, it works as a two-state latch circuit. C.S.Lent shows that a cellular logic circuits can be constructed by using the quantum cells, where Coulombic interaction is used for interconnection instead of the wired interconnection[Lent *et al.* 1993].

A fundamental logic (e.g. OR circuit) is shown as in Figure 6. For realization of Lent's circuit, we will have the following problems:

1) Can Coulombic interaction work really as the interconnection ?
2) Is it appropriate for realization of the conventional logic ?

The first problem must be solved for both interconnection within a quantum cell and interconnection among several quantum cells. We expect that the first problem will be solved by q.f.d. researchers, and refer to the second problem hereafter.

We have constructed a prototype of design tool where a logic function is automatically transformed into a Lent's circuit, and have tried to generate several Lent's circuits. As a result we have faced a problem that two-dimensional realization of Lent's circuit will require a lot of space for the general logic circuit, since the space of interconnection becomes larger than we expected. An example is shown as in Figure 7. The prototype tool is not complete for minimization of space, but we will have the space overhead for

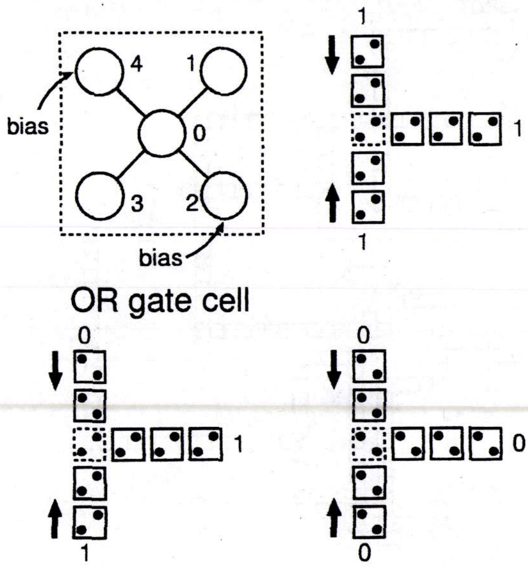


Fig. 6. OR Logic Circuit.

realization of Lent's circuit. Even if Coulombic interaction reduces the interconnection space, a certain amount of space for interconnection will be required and it will be much larger than we expect. The overhead for quantum logic circuit will become really the system constraint, since the system is simple and the system constraint coincides with the interconnection overhead in this case.

When considering the interconnection overhead, we propose a realistic design strategy as follows;

1) Quantum cells are used only for memories.

2) Logic circuit to access the memories is realized by the conventional CMOS circuit.

We will have still a problem how to realize an access circuit to memories, but will have an appropriate circuit if the latch circuit is realized by a quantum cell. Of course, the access circuit using CMOS circuit becomes enough large as a total system, but will fit an FPGA(Field Programmable Gate Array) in RAM(Random Access Memory)-type.

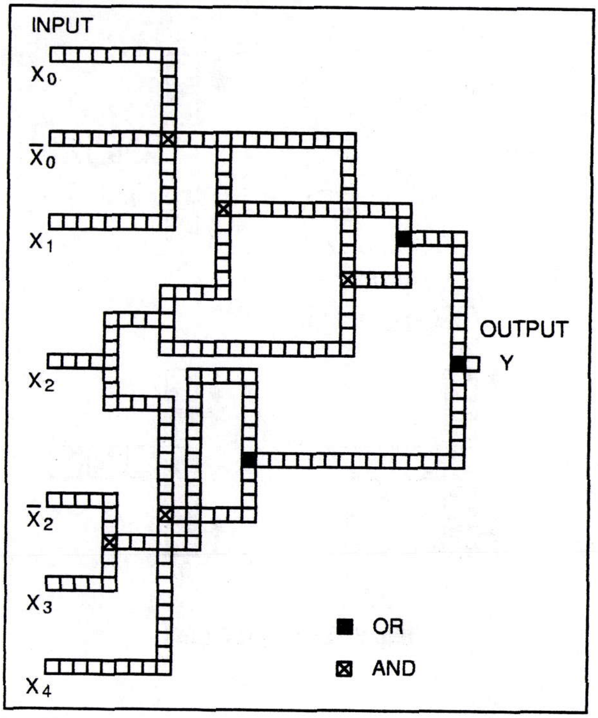


Fig. 7. Lent's Circuit of $Y = X_0 \cdot X_1 \cdot X_2 + \bar{X}_0 \cdot X_2 + \bar{X}_2 \cdot X_3 + X_2 \cdot X_4$.

4 Quantum Computing

Research for quantum functional device and research for quantum computing model are usually regarded as being independent each. We should notice that the quantum computing is also applicable for q.f.d. research. The quantum behavior should be included as a feature of q.f.d. system, since the q.f.d. has a constraint for realization due to quantum behavior.

We will introduce here the quantum computing briefly. (See e.g.[Hosoya 1999] for the details.)

A typical example of quantum computing is the quantum Turing machine (sometimes, quantum computer) used by P.W.Shor, where he shows an algorithm for factorization of integers with ultra high speed, but with a very low error rate[Shor 1994]. The quantum gate with the function of Controlled NOT is introduced for construction of quantum Turing machine, and the realization of Controlled NOT has been often discussed after his proposal. The realization of Shor's tricky algorithm has a difficulty, and we need to make much more efforts to realize his circuit without loss of high speed computation, since the algorithm must be implemented as a quantum computation.

From the viewpoint of more q.f.d. site we give up to realize the quantum gate (the logic circuit part), and focus on only realization of the quantum state, which will bring us a kind of merit of q.f.d. application. One of applications of quantum computing is known to be applicable for cryptography, where it takes a huge time (really, an exponential time) to decode a cryptogram. For instance, there exist 2^n for n-dimensional binary vector whose component is 0 or 1 each, and the selection among 2^n candidates will require an exponential time.

Example : Case of $n = 4$, we will have 16 candidates as follows;
$[0000], [0001], [0010], [0011], \ldots, [1110], [1111]$.

Though we will have a solution quickly for the case of $n = 4$, we cannot obtain correctly a solution for the case of $n = 40$ or more, even with a highly parallel supercomputer. The exponential complexity is utilized for a cryptogram, since it cannot be really decoded by an another person except the sender and the receiver.

When comparing the design of the quantum Turing machine with the design of the conventional computer, we will have

Quantum State: Data
Control of Quantum State: Control,

and therefore, we focus on only realization of the quantum state. It will be more realistic to give up the realization of the control part. We call the quantum state the *quantum memory*, since it will be realized as a memory within a system.

The feature of quantum memory is that each component includes an entangled state, where it is represented as a complex number, though a component of the conventional memory has only 0 or 1, i.e., a binary state. The quantum Turing machine is not directly related to the q.f.d., but the realization of quantum state, i.e., entangled state will be a purpose of q.f.d. application.

The quantum state (qubit) is represented as follows:
$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \ where \ |\alpha|^2 + |\beta|^2 = 1.$

The complex number means an entangled state, which has a more various state than the conventional binary state. The realization of qubit is not easy problem, and several trials are continued among quantum computer researchers. It will be also for goal for q.f.d. researchers[Fujisawa *et al.* 2000].

5 Toward Quantum Memory

We must consider how to realize the quantum state, since the conventional n-bit memory is not quantum memory even if each bit is realized by q.f.d' s, i.e., a cell of quantum dots or an SET(Single Electron Transistor) flip-flop.

It is necessarily required to utilize the interaction among q.f.d. bits, which consist of a one-bit memory realized by q.f.d.'s. We focus on two types of q.f.d.'s, i.e., a quantum dot and an SET.

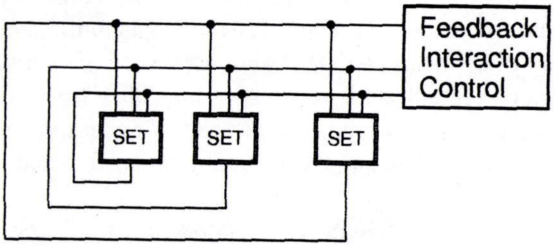


Fig. 8. An Idea for Quantum State Memory.

5.1 Interaction among Quantum Dots

The circuit is a correction of quantum dots just like Lent' quantum cell. The space structure, however, should be three-dimensional since more than ten bit quantum dots must be equally interconnected with each other by electrostatic interaction. Moreover, it is also required to access each quantum dot for read/write operation. We conclude that the realization using quantum dots will be too difficult and that it will be not realizable.

5.2 Interaction among SETs

The circuit proposed by Amemiya's Hopfield Network will be an important key idea to realize a quantum state memory. The Hopfield Network is considered to be useful to solve an NP-complete problem, i.e.. a problem to require an exponential time for solving. Amemiya's circuit is realized by cooperative tunneling on the SET Hopfield Network, and the simulation shows that the problem will be solved without local minima[Akazawa *et al.* 2000].

We cannot divide the operation part (to obtain the result) and the data part(to maintain the bit memory) for the Hopfield Network, since the network is an indivisible structure. For the quantum state memory we construct only the data part(the memory part) and maintain the memory as a quantum state memory, where each bit must be qubit. The idea of a circuit is shown as in Figure 8, where the amount of feedback interaction will be very important.

Tuning of feedback interaction must satisfy interaction and tunneling. If the interaction is too weak, the cooperative tunneling will disappear, and each bit will be the conventional binary bit (not qubit). Moreover, we need to investigate the algorithm using the quantum state memory, since the tunneling is a probabilistic behavior, though the circuit for algorithm implementation is supposed to be constructed by a conventional CMOS circuit.

6 Conclusion

For q.f.d.'s (quantum functional devices) we must have a goal to realize an original circuit, not one of conventional CMOS circuits. The quantum computer is regarded as an another research for q.f.d.'s, but we should discuss a wide concept including it.

We proposed an idea to develop a q.f.d. circuit, but it is not the same as the quantum computer. From the viewpoint of computer design, we should realize the control part not by q.f.d.'s, but by the conventional CMOS circuit, and make efforts to realize the data part, i.e., the quantum state memory. As a result we have a hybrid computer as a goal.

We need to investigate not only the realization of quantum state memory, but also the quantum algorithm, i.e., the algorithm using the quantum state memory. The realization of such a hybrid computer will give us a great impact, though the research is rather difficult, and will require a large time (e.g., a decade).

References

[Lent *et al.* 1993] C.S.Lent, P.D.Tougaw, W.Porod, and G.H.Bernstein, "Quantum Cellular Automata", Nanotechnology 4, pp.49–57 (1993).

[Hosoya 1999] A.Hosoya, Lectures on Quantum Computation, Science-sha, (1999, in Japanese).

[Shor 1994] P.W.Shor, "Algorithm for quantum computation:Discrete logarithms and factoring", Proc. 35th IEEE Symp. on FOCS, pp.124–134 (1994).

[Fujisawa *et al.* 2000] T. Fujisawa and Y. Hirayama, "Transmission type rf single electron transistor operation of a semiconductor quantum dot", Jpn. J. Appl. Phys. vol.39, pp.2338–2340 (2000).

[Akazawa *et al.* 2000] M.Akazawa, E.Tokuda, N.Asahi and Y.Amemiya, "Quantum Hopfield Network using single-electron circuits – A novel Hopfield Network free from local-minimum difficulty", Analog Integrated Circuits and Signal Processing, vol.24, pp.51–57 (2000).