January 24, 2006

Physics 681-481; CS 483: Assignment #1

(please hand in after the lecture, Thursday, February 2)

These assignments (which will appear every other week) have three purposes. They explore points not covered in the lectures or lecture notes; they give you practice manipulating the formalism developed in the lectures and the notes (in the case of this assignment, Chapter 1); and they give you an opportunity to produce some *clear technical writing.*

In every case after you've done the analysis to answer the questions you should write a brief essay reporting the results of that analysis at the expository level of the lecture notes — i.e. your report should be presented in sentences, organized into paragraphs, and mathematical details that competent readers can construct for themselves should be omitted. Your goal should be to explain the answers as clearly as you can to somebody else in the class who was puzzled by the question, but doesn't need to be led by the hand through obvious details.

What you should try to produce is additional text that could have been added to the lecture notes. After the assignments are handed in I'll post my own attempts. If you've done a good job you should prefer yours to mine. Hand in only the finished text; not your working papers. After you see my own first attempt, you should have a better idea of what I'm looking for.

Apologies if you find this first assignment irritatingly technical or boringly straightforward. I want to make sure that people who are not acquainted with this formalism acquire the familiarity with it they can only get by manipulating symbols. The challenge for those who are already familiar will be to find a graceful way of explaining things. Subsequent assignments should be more entertaining.

I. Tensor products and positional notation.

Section A of Chapter 1 shows that if an integer x between 0 and $2^n - 1$ is represented by a 2^n-component column vector with all components 0 except for a 1 in the position x places down from the top place (the top place is 0 places down from itself), then if the number is represented in binary, that column vector is just the tensor product of the n 2-component column vectors that represent the values of its n bits. Give an argument that this is true of decimal numbers as well, taking as your example, if you wish, the number 532, showing that its representation as a 1000-component column vector (all 0's except for a 1, 532 places down from the top place) is just the tensor product of the three 10-component column vectors representing (from right to left) the digits 2, 3, and 5. Since it is hard to fit a 1000 entry column vector on the page, this requires an explanation that is, at least in part, verbal.

II. Manipulating elementary operators.

Carry out the "more algebraic" derivation mentioned at the top of page 13 of Chapter 1 of the relation

$$\mathsf{S}_{ij} = \mathsf{C}_{ij}\mathsf{C}_{j\,i}\mathsf{C}_{ij}, \tag{1}$$

between the SWAP operator in the form

$$\mathbf{S}_{ij} = \mathbf{n}_i\mathbf{n}_j + \overline{\mathbf{n}}_i\overline{\mathbf{n}}_j + (\mathbf{X}_i\mathbf{X}_j)(\mathbf{n}_i\overline{\mathbf{n}}_j + \overline{\mathbf{n}}_i\mathbf{n}_j), \tag{2}$$

and the cNOT operator in the form

$$\mathbf{C}_{ij} = \overline{\mathbf{n}}_i + \mathbf{X}_j\mathbf{n}_i. \tag{3}$$

You should use only the identities

$$\mathbf{n}^2 = \mathbf{n}, \quad \overline{\mathbf{n}}^2 = \overline{\mathbf{n}}, \tag{4}$$

$$\mathbf{n}\overline{\mathbf{n}} = \overline{\mathbf{n}}\mathbf{n} = \mathbf{0}, \tag{5}$$

$$\mathbf{n} + \overline{\mathbf{n}} = \mathbf{1}, \tag{6}$$

$$\mathbf{n}\mathbf{X} = \mathbf{X}\overline{\mathbf{n}}, \quad \overline{\mathbf{n}}\mathbf{X} = \mathbf{X}\mathbf{n}, \tag{7}$$

$$\mathbf{X}^2 = \mathbf{1}, \tag{8}$$

obeyed by the 1-bit NOT and number operators, and the fact that 1-bit operators acting on different Cbits commute.

III. The Toffoli gate.

The 3-Cbit controlled-controlled-NOT gate $\mathbf{T}_{ijk}$ (also called the Toffoli gate, after its inventor) plays a very important role in the classical theory of reversible computation. It has two control bits (i and j) and a target bit (k), and is defined to act as the identity unless both control bits are 1, in which case it acts as NOT on the target bit.

Find the 8×8 matrices in the set of 3-Cbit states $|x_2\rangle|x_1\rangle|x_0\rangle$ of $\mathbf{T}_{210}$, $\mathbf{T}_{201}$, and $\mathbf{T}_{102}$ by generalizing to 3-Cbit states the procedure used to construct the 4×4 matrices of cNOT on pages 14 and 15 of chapter 1.