Lecturer: Yongshan Ding *Scribe: Reza Jokar*

Lecture 9: Memory management

November 6, 2018

1 Introduction to Quantum Memory Management

In classical computers, memory management is typically done using a memory hierarchy. Memory hierarchy includes different layers of memories including CPU registers, different levels of caches, main memory and disk. As we go to upper levels in the memory hierarchy, the memory becomes smaller and more expensive, as shown in Figure 1. In quantum computers, on the other hand, there is no memory hierarchy because the concept of memory hierarchy relies on the fact that we can read/write and copy the data efficiently, which is not the case in quantum computers. For example, we know that read operation is done using measurement which destroys the state of the qubits. We discuss how to copy the value of qubits later in this lecture. Note that in this lecture the focus is NISQ machines that are not large and we do not really need memory hierarchy, and we are in the CPU registers level. There are some research papers that introduce the concept of memory hierarchy for large quantum computers that use teleportation to copy data between memory levels.

2 Copying a Qubit State

Let us assume we have a qubit in an arbitrary state of $\alpha|0\rangle + \beta|1\rangle = \begin{pmatrix}\alpha\\ \beta\end{pmatrix}$. Here, we discuss performing copy operation using EPR pairs. EPR pair is in the state of $\frac{1}{\sqrt{2}}(\alpha|00\rangle + \beta|11\rangle)$ including two quantum bits that are entangled. The two qubits are also identical and when Alice and Bob each has one of the qubits:

- if Alice measure the qubit and reads $|0\rangle$, then Bob's qubits also collapse to $|0\rangle$
- if Alice measure the qubit in X-basis and reads $|+\rangle$, then Bob's qubits also collapse to $|+\rangle$

The qubits in EPR pair are the same regardless of the basis and are truly identical; in some sense, they are copies of each other. Thus, we can create an identical pair in the $|+\rangle$ and $|0\rangle$ states. But can we create an identical pair of any arbitrary state of $\begin{pmatrix}\alpha\\ \beta\end{pmatrix}$? The answer is yes, however, we still cannot manipulate each copy similar to classical computers because the qubits are entangled. We demonstrate an example to show how entanglement

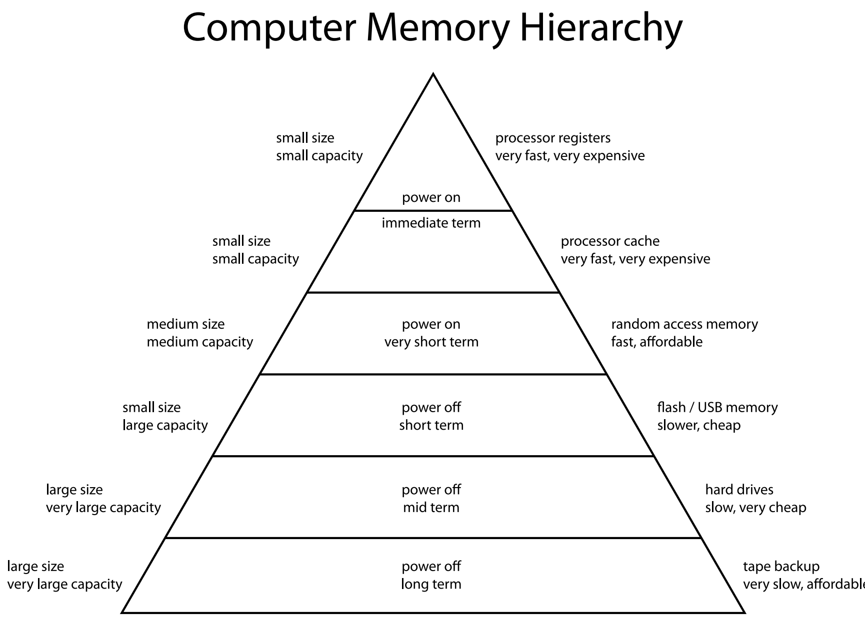

Figure 1: Memory Hierarchy in Classical Computers [Wik].

can kill the interference ability. For example if a qubit is in the $|+\rangle$ state and we perform a *Hadamard* gate on that qubit before the copy, the output is $\frac{1}{2}(\alpha|0\rangle+\beta|1\rangle)+\frac{1}{2}(\alpha|0\rangle-\beta|1\rangle) = |0\rangle$; however, if we apply the *Hadamard* gate on the qubit after the copy, the output is $\frac{1}{2}(\alpha|00\rangle+\beta|11\rangle)+\frac{1}{2}(\alpha|00\rangle-\beta|11\rangle)$, and we cannot cancel the amplitudes of the first qubit. Thus, ideally, we need a quantum pair which are identical but are not entangled.

Figure 2: Copying any Arbitrary Quantum State.

Next, we want to see if we can find a unitary U that copies any arbitrary state. Let us assume we have a device that implement the unitary U, and we can a quantum bit in an arbitrary state of $\psi = \alpha|0\rangle + \beta|1\rangle$; then, we need to have two copies of the qubit as output, and the output is $\psi \otimes \psi$. As shown in Figure 2, we also introduce $|0\rangle)$ as the second input to the device that implemented the unitary U. We need the device to convert the input that is

$\psi \otimes |0\rangle = (\alpha|0\rangle + \beta|1\rangle) \otimes |0\rangle$ to $(\alpha|0\rangle + \beta|1\rangle) \otimes (\alpha|0\rangle + \beta|1\rangle) = \alpha^2|00\rangle + \alpha\beta|01\rangle + \beta\alpha|10\rangle + \beta^2|11\rangle$. The input to output mapping is not linear and we cannot find a matrix that transform the input vector to output vector. Note that based on the transformation rule, all quantum transformations have to be linear, and it is not possible to have such mapping. Thus, we cannot create an unentangled identical pair of quantum bits and it was discovered in [WZ82] as no-cloning theorem which is defined as below:

- No-cloning theorem: there is no unitary U that can transform all qubit stats $\psi = \alpha|0\rangle + \beta|1\rangle$ to $\psi \otimes \psi$.

Note that the theorem says that we cannot transform all qubit states, however, it is possible to transform particular known states. For example, it is possible to transform $|0\rangle$ to $|0\rangle \otimes |0\rangle$, or $|+\rangle$ to $|+\rangle \otimes |+\rangle$. Basically, no-cloning theorem says that there is no single unitary that can transform all qubit states. Note that no-cloning has strong implications on error correction schemes because in traditional fault tolerant systems we need to have a copy of the data, but we cannot have that here.

3 Minimizing the Number of Qubits in the Computation

As we cannot copy the qubits and qubits are scarce, the next question is how do we minimize the number of qubits we use in the computation. We first need to demonstrate reversible computation:

- Def: A function is reversible if $f : \{0,1\}^n \to \{0,1\}^n$, and it has unique input for each output, so that we can recover the inputs from the outputs.

Examples of classical reversible gates are *Not*, *CNOT* (*XOR* in classical computers), *Toffoli*, *Swap* and *Fredkin* gates. Note that *Toffoli* and *Fredkin* gates are universal and any reversible computation can be done using only *Toffoli* and *Fredkin* gates.

- Remark: Reversible logic gates can be regarded as permutations of the bit strings. For example, *Not* gate transform $\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ to $\begin{pmatrix} \beta \\ \alpha \end{pmatrix}$ and *CNOT* gate transform $\begin{pmatrix} \alpha 1 \\ \alpha 2 \\ \alpha 3 \\ \alpha 4 \end{pmatrix}$ to $\begin{pmatrix} \alpha 1 \\ \alpha 2 \\ \alpha 4 \\ \alpha 3 \end{pmatrix}$. Note that there is no creation or annihilation of superpositions. In other words, we cannot map non-zero entries to zero, and the number of zero entries in the input vector is equal to the number of zero entries in the output vector. We can see that hadamard gate is not a classical reversible gate because it maps $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ to $\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}$ and that is an example of creation of superposition.

3.1 Ancilla

Next, we discuss the implementation of *CCCNOT* ($\Lambda^3(x)$) gates, and usage of ancilla and dirty ancilla. We use *NOT*, *CNOT*, *Toffoli* gates, and ancillary/scratch qubits to implement *CCCNOT* as shown in Figure 3. Figure 3 shows that we use an ancilla qubit in the 0 state. The state of the ancilla qubit is changed after performing the gate operation and a garbage value is generated. Garbage needs to be uncomputed and one easy way to uncompute the garbage is by flipping the bit back using another symmetric *Toffoli* gate; now, the state of the ancilla qubit is flipped back to 0.

To demonstrate dirty ancilla, we can continue and add another *Toffoli* gate as shown in Figure 4. By doing so, we relax the constraint on the state of the ancilla qubit, and it does not necessarily need to be in 0 state anymore; the ancilla qubit can be in any arbitrary state and the state is returned to the original value after performing the gate operation. We call this ancilla, dirty ancilla; it can borrowed from anywhere on the chip to help in performing the *CCCNOT* computation, and it is returned (and its state is restored to the original value) after the gate operation is done. Note that uncomputation is not just a circuit optimization, and in many cases, it is necessary for the correctness of the computation. For example, we observed in the EPR pair demonstration that if we have some entangled garbage coupled with the output, we will not be able to accomplish interference on the output qubit. Basically, without the uncomputation, the interference ability is disabled.

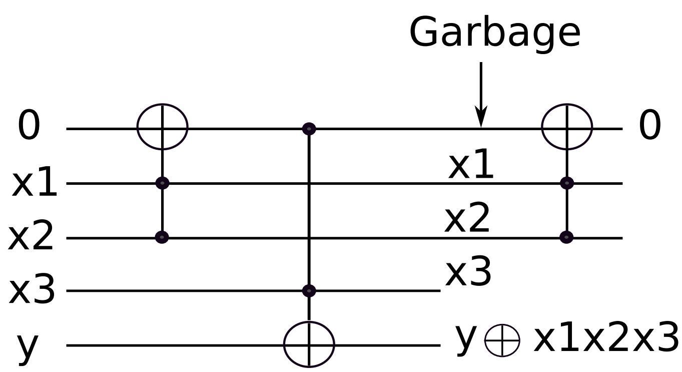

Figure 3: Implementation of *CCCNOT* Gates using an Ancilla Qubit in the 0 State.

3.2 Uncomputation Trick

Uncomputation is performed in order to get rid of garbage and is invented by Bennet in 1989. The uncomputation trick is done as follow:

- First: we run the computation forward. When running the circuit forward, we accumulate the junk. We do not touch the junk and at the end, it will be in an unknown state.

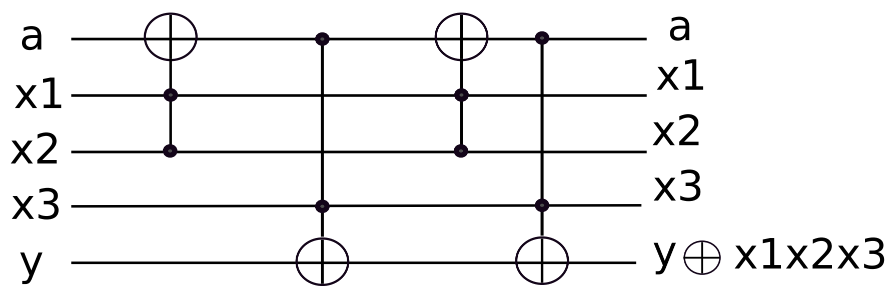

Figure 4: Implementation of *CCCNOT* Gates using an Ancilla Qubit in an Arbitrary State.

- Second: we copy the output to a *safe* place. As shown in Figure 5 we copy the output to a clean ancilla.
- Third: we run the computation backward by executing the gates in reverse order. Each gate is replaced with its own inverse.

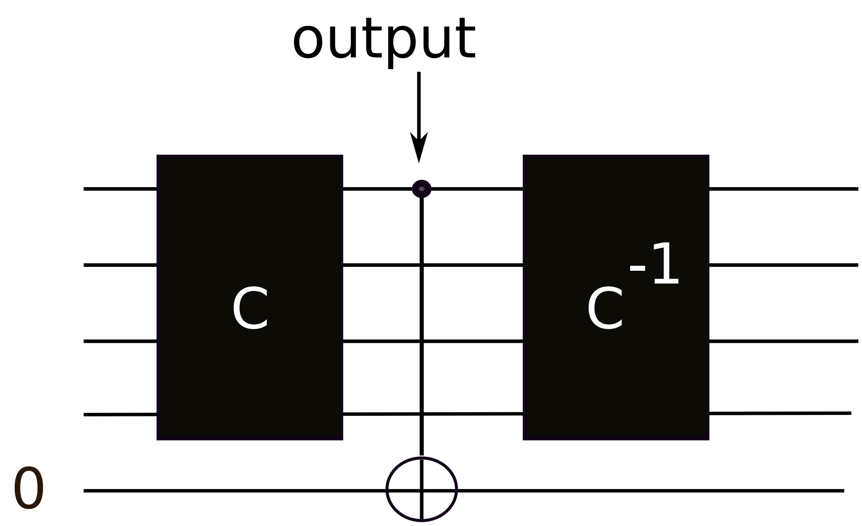

Figure 5: Uncomputation Trick.

As we discussed earlier copying using *CNOT* gates is not what we want because the two copies are identical but also entangled. But why does it work for us in this case? Does it work for all the circuits? And does it work for quantum states? These are the questions that we will answer next. We need to emphasize some key points in order to answer these questions. The key points are:

- Uncomputation trick works only if the circuit C is a classical reversible circuit. In other words, we cannot have *Hadamard* gate in the circuit.

- Uncomputation trick works for quantum state input (i.e., superposition is fine). We discussed that classical reversible circuits can be regarded as a permutation of bit strings. For example, let us assume that circuit C maps *00* to *10*, *01* to *00*, *10* to *01* and *11* to *11*. The input state is $\psi = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \sigma|11\rangle$. The state changes to $\alpha|10\rangle + \beta|00\rangle + \gamma|01\rangle + \sigma|11\rangle$ after the circuit C. After performing the $CNOT$ gate, we will have a 3 qubit system and the state is $\alpha|101\rangle + \beta|000\rangle + \gamma|010\rangle + \sigma|111\rangle$. Finally, the state after applying the C^{-1} is $\alpha|001\rangle + \beta|010\rangle + \gamma|100\rangle + \sigma|111\rangle$. We achieved what we wanted, and the circuits works for quantum state inputs as long as C is a permutation without superposition creation and annihilation.

References

[Wik] Wikipedia. Memory hierarchy. [Online; accessed 10-December-2018].

[WZ82] WK Wooters and WK Zurek. Quantum no-cloning theorem. *Nature*, 299:802, 1982.