

Building Blocks for Quantum Computing PART II

Patrick Dreher
CSC801 – Seminar on Quantum Computing
Spring 2018

Building Quantum Circuit Gates Using Qubits and Quantum Mechanics

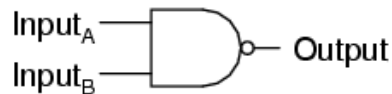
Classical Gates versus Quantum Gates

- Quantum physics puts restrictions on the types of gates that can be incorporated into a quantum computer
- The requirements that
 - A quantum gate must incorporate the linear superposition of pure states that includes a phase
 - all closed quantum state transformations must be reversiblerestrict the type of logic gates available for constructing a quantum computer
- The classical NOT gate is reversible but the AND, OR and NAND gates are not

Classical Logic Gates

- There are several well known logic gates

NAND gate



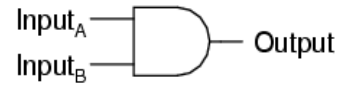
A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0

Exclusive-OR gate



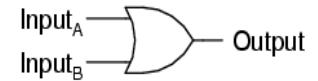
A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

AND gate



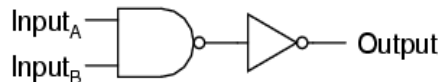
A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1

OR gate

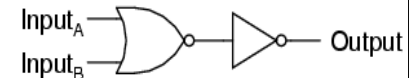


A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

Equivalent circuit




Equivalent circuit





Quantum Property of Reversibility and Constraints of Gate Operations


- Reversibility can be quantified mathematically through the matrix representation of the logic gate
- The *IDENTITY* operation and *NOT* gates are “reversible” the classical NOT gate are reversible, the AND, OR and NAND gates are not
- The outcome of the gate can be undone by applying other gates, or effectively additional matrix operations
- The matrix has the property of preserving the length of vectors, implying that the matrices are unitary, thereby satisfying the Axiom 4 requirement for quantum mechanics
- For gates represented by a matrix, the unitarity condition is necessary and sufficient for ensuring that pure states get mapped to pure states


Recall Symbols for Single Qubit Gates


Pauli X  $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \sigma_x$

Pauli Y  $\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} = \sigma_y$

Pauli Z  $\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = \sigma_z$

Phase  $\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$

$\frac{\pi}{8}$  $\begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}$

Hadamard  $\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

Pauli Spin Matrices*

* Pauli spin matrices give a hint toward potential designs for a building a quantum computer

Recall Rotation Gates

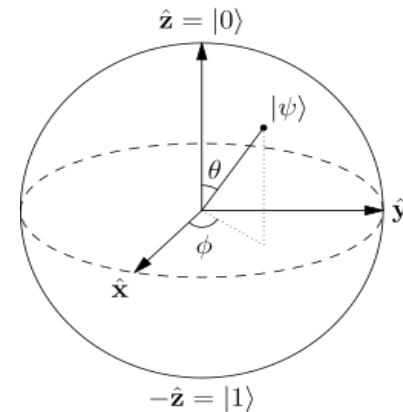
- Use the identity

$$R_{\hat{n}}(\theta) \equiv e^{-i\theta\hat{n}\cdot\frac{\sigma}{2}} = \cos\left(\frac{\theta}{2}\right)I - i\sin\left(\frac{\theta}{2}\right)(n_xX + n_yY + n_zZ)$$

- The R gate can specify a rotation in a specific direction by a specific angle

example

$$R_y(\pi/4)$$



Reversibility for Multi-Qubit Gates

- The output of the 1 bit NOT gate can be reversed by applying another NOT gate
- Construct a 2 qubit gate that satisfies the reversibility condition (i.e. gate needs to be represented by a unitary matrix)

Reversible 2 Qubit CNOT Gate

- A two qubit quantum logic gate has a control qubit and a target qubit
- The gate is designed such that if
 - the control bit is set to 0 the target bit is unchanged
 - The control bit is set to 1 the target qubit is flipped

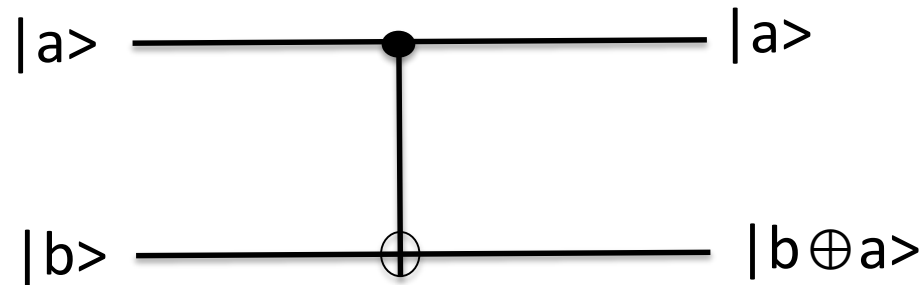
Input	Output
$ 00\rangle$	$ 00\rangle$
$ 01\rangle$	$ 01\rangle$
$ 10\rangle$	$ 11\rangle$
$ 11\rangle$	$ 10\rangle$

- Can be expressed as $|a, b\rangle \longrightarrow |a, b \oplus a\rangle$
- The CNOT gate is generally used in quantum computing to generate entangled states

Controlled-NOT Gate

Matrix representation of the CNOT gate

$$U_{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad U_{CNOT}^\dagger U_{CNOT} = I$$

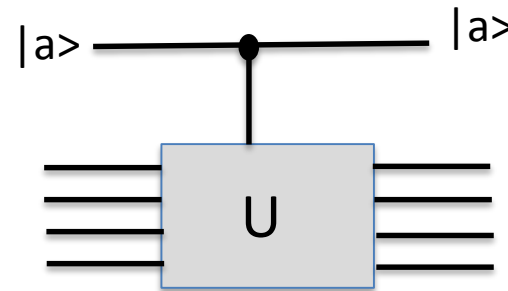


$$|a\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad |b\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \begin{array}{l} |aa\rangle \rightarrow |aa\rangle \\ |ab\rangle \rightarrow |ab\rangle \end{array} \quad \begin{array}{l} |ba\rangle \rightarrow |bb\rangle \\ |bb\rangle \rightarrow |ba\rangle \end{array}$$

Homework – Construct the U_{CNOT} from the rules for building a CNOT gate and show that this gate is reversible and therefore a good quantum gate

Controlled U Gate

- Extension of the controlled CNOT gate
- Given any unitary matrix U can construct a universal gate with the properties
 - Single control qubit
 - N target qubits
- Outputs
 - If the control bit is set to “0” the target bits are unchanged
 - If the control bit is set to “1” then the gate U is applied to the target bits



Other Controlled Gates

- Controlled U gate is a gate that operates on two qubits in such a way that the first qubit serves as a control. It maps the basis states as follows

$$|00\rangle \rightarrow |00\rangle$$

$$|01\rangle \rightarrow |01\rangle$$

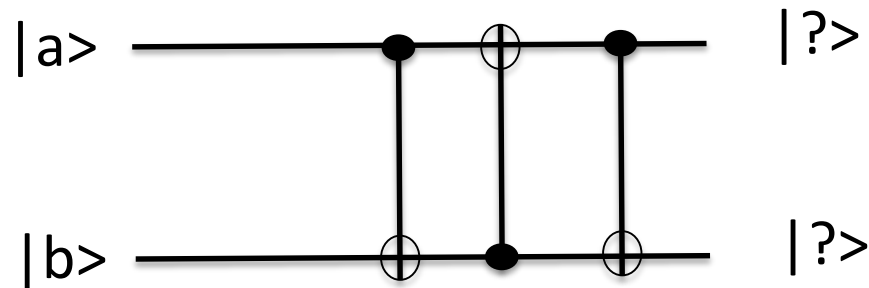
$$|10\rangle \rightarrow |1\rangle \otimes U|0\rangle = |1\rangle \otimes (u_{00}|0\rangle + u_{10}|1\rangle)$$

$$|11\rangle \rightarrow |1\rangle \otimes U|1\rangle = |1\rangle \otimes (u_{01}|0\rangle + u_{11}|1\rangle)$$

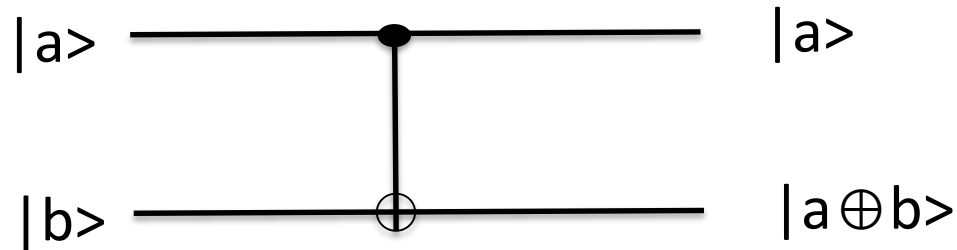
$$C(U) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{00} & u_{01} \\ 0 & 0 & u_{10} & u_{11} \end{pmatrix}$$

- U represents one of the Pauli matrices σ_x σ_y σ_z
- Controlled-X, Controlled-Y, Controlled-Z gates

What Happens When Multiple CNOT Gates Are Combined?

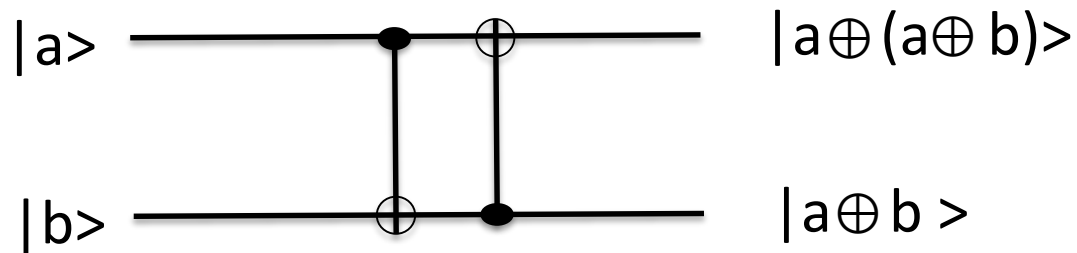


Combine Multiple CNOT Gates



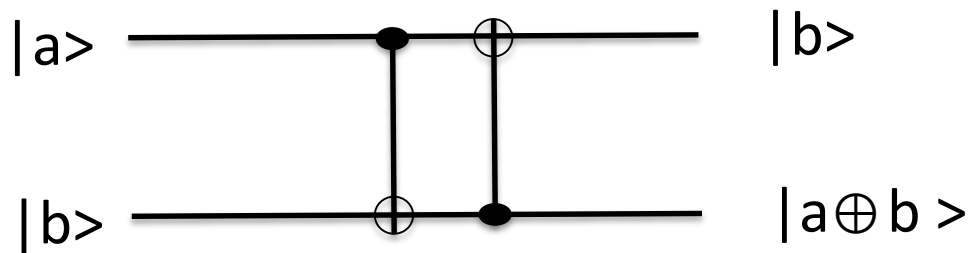
a	b	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

Combine Multiple CNOT Gates



a	b	$a \oplus b$	a	$a \oplus (a \oplus b)$
0	0	0	0	0
0	1	1	0	1
1	0	1	1	0
1	1	0	1	1

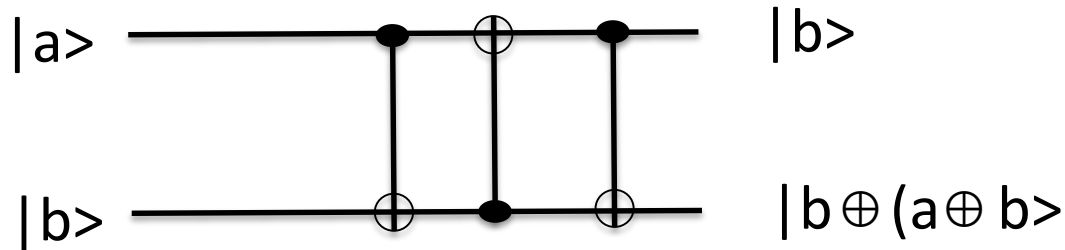
Combine Multiple CNOT Gates



a	b	$a \oplus b$	a	b
0	0	0	0	0
0	1	1	0	1
1	0	1	1	0
1	1	0	1	1

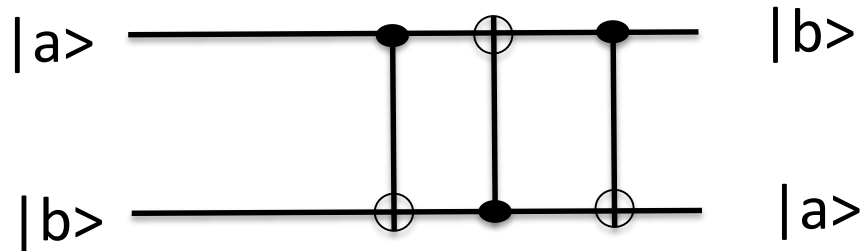


Combine Multiple CNOT Gates



a	b	$a \oplus b$	a	b	$a \oplus b$	$b \oplus (a \oplus b)$
0	0	0	0	0	0	0
0	1	1	0	1	1	0
1	0	1	1	0	1	1
1	1	0	1	1	1	1

SWAP Gate



a	b	$a \oplus b$	a	$a \oplus (a \oplus b)$	$a \oplus b$	a
0	0	0	0	0	0	0
0	1	1	0	1	1	0
1	0	1	1	0	1	1
1	1	0	1	1	1	1

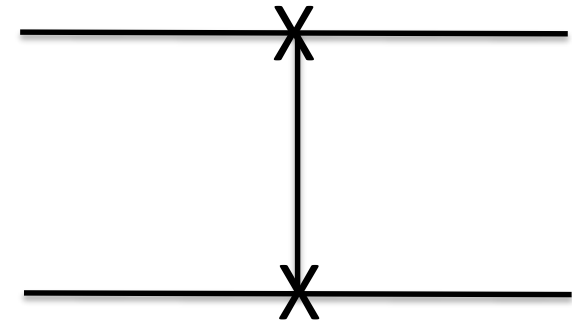


Matrix Representation of the SWAP Gate

Truth Table for the SWAP Gate

Input	Output
$ 00\rangle$	$ 00\rangle$
$ 01\rangle$	$ 10\rangle$
$ 10\rangle$	$ 01\rangle$
$ 11\rangle$	$ 11\rangle$

SWAP Gate circuit representation

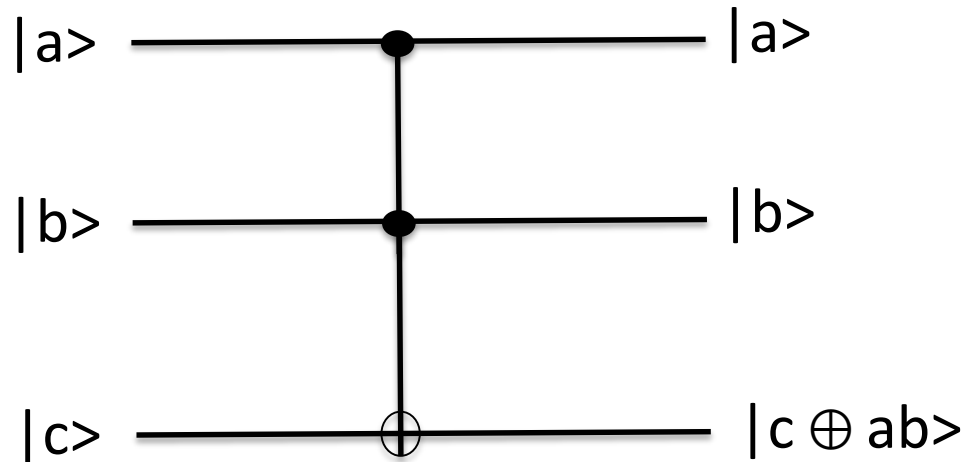


$$U_{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Homework – Construct the U_{SWAP} from the rules for building a SWAP gate and show that this gate is reversible and therefore a good quantum gate

Toffoli Gate

- The Toffoli gate is a 3-bit gate, which is universal for classical computation
- If the first two bits are in the state $|1\rangle$, it applies a Pauli-X (NOT) on the third bit, otherwise the state is left unchanged

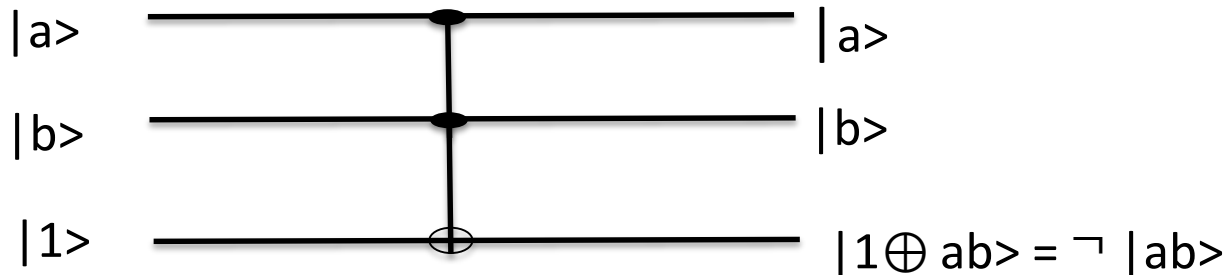


Properties of Toffoli Gates

- Toffoli Gate is a reversible gate (i.e. $U_T^{-1}U_T=I$) or
- Toffoli gate is used to replace a classical circuit with the equivalent reversible gate
- Two bits are control bits ($|a\rangle$ and $|b\rangle$) and target bit $|c\rangle$ is flipped as per the truth table

$$(a, b, c) \rightarrow (a, b, c \oplus ab) \rightarrow (a, b, c)$$

- Toffoli gate can be used to simulate a NAND Gate



Toffoli Gate Truth Table and Matrix

INPUT			OUTPUT		
a	b	c	a'	b'	c'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

$$\begin{pmatrix}
 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
 \end{pmatrix}$$

X Gate
Pauli σ_x rotation matrix

Toffoli Gate as a Universal Gate

- A Toffoli gate constructs the AND logic state when $c = 0$
- A Toffoli gate constructs the NAND when $c = 1$
- Every Boolean function has a reversible implementation using Toffoli gates
- There is no universal reversible gate with fewer than three inputs
-

A Reversible Universal Logic Gate

- A controlled SWAP can be defined as

$$F = |0\rangle\langle 0| \otimes \mathbb{1} \otimes \mathbb{1} + |1\rangle\langle 1| \otimes S$$

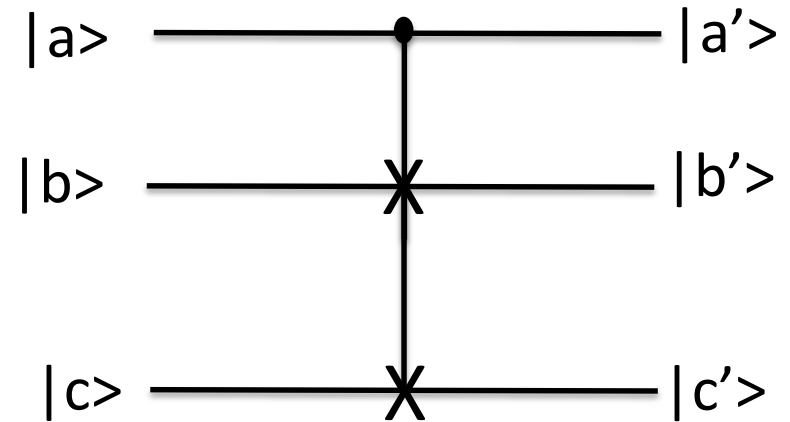
where S is the usual swap operation

$$S = |00\rangle\langle 00| + |01\rangle\langle 01| = |10\rangle\langle 10| + |11\rangle\langle 11|$$

- The number of 1s is conserved between the input and output (conservative reversible logic gate)
- This reversible universal logic gate can be constructed as a 3-bit gate that performs a controlled swap

Fredkin Gate (CSWAP) Properties

- Property that the $|c\rangle$ is the control bit and is not changed by the Fredkin gate
- If $|c\rangle=0$ then $|a\rangle$ and $|b\rangle$ are unchanged
- If $|c\rangle=1$ then $|a\rangle$ and $|b\rangle$ are swapped
- The original Fredkin Gate settings can be recovered by applying the Fredkin gate twice



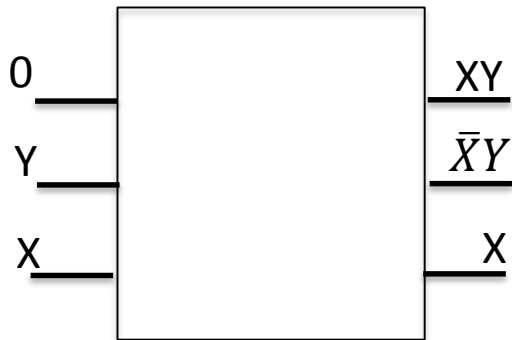
Fredkin Gate Truth Table and Matrix

INPUT			OUTPUT		
a	b	c	a'	b'	c'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	0	1
1	1	1	1	1	1

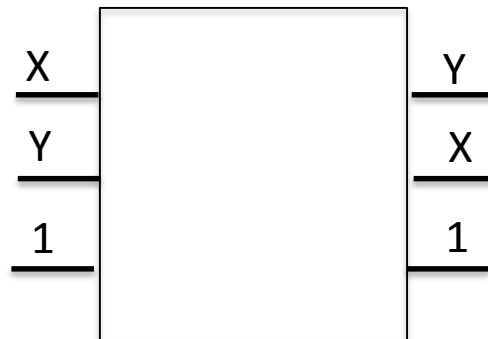
$$\begin{pmatrix}
 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
 \end{pmatrix}$$

X Gate
Pauli σ_x rotation matrix

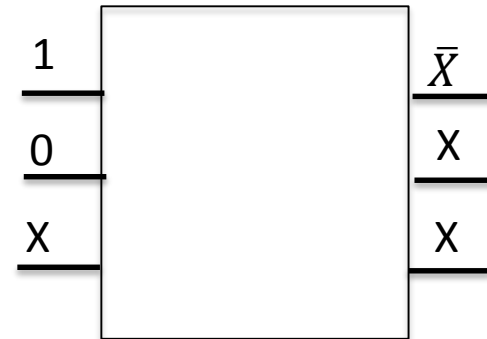
Fredkin Gates Mapping Classically Irreversible Gates



AND Gate

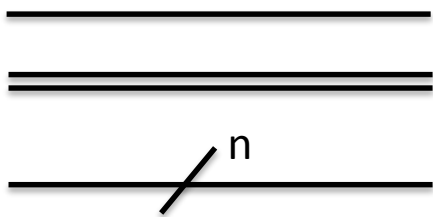


Crossover Gate



NOT Gate

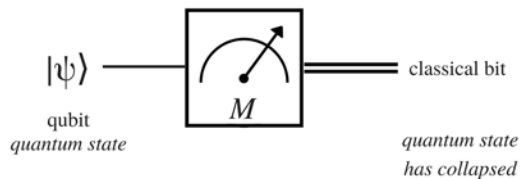
Other Diagrams and Circuit Symbol Nomenclature for Quantum Gates



Single qubit propagating forward in time

Classical bit propagating forward in time

n qubits propagating forward in time



Measurement projection onto pure states $|0\rangle$ and $|1\rangle$

Other gates

- Single qubit gates already illustrated
- Additional multi-qubit gates

Summary -- Gates

- Any quantum gate that is used to construct quantum computing operations must have a truth table that preserves the following
 - The gates must operate in a complex vector space
 - Complex vector space linear transformations that preserve orthogonality are unitary transformations
 - The dynamics that takes states from t_1 to t_2 are restricted to transformations that preserve this orthogonality and are therefore represented by unitary matrices

Quantum Circuits

- A quantum circuit is a sequence of gates connected by “wires” (qubit propagating forward in time)
- A quantum circuit is limited fixed width that corresponds to the number of qubits being processed
- Goal is to construct a circuit structure that
 - conforms to the postulates of quantum mechanics
 - Independent of physical technology
 - Functionally correct

Quantum Circuits

- The 1 qubit rotation gates plus the universal gates that have been discussed form the “lego building blocks” for constructing arbitrarily complex quantum circuits
- Needs higher level of care than just randomly picking and combining rotation gates and universal gates
- Each proposed quantum circuit must be validated that
 - It satisfies the properties of quantum mechanics
 - Produces the correct functionality

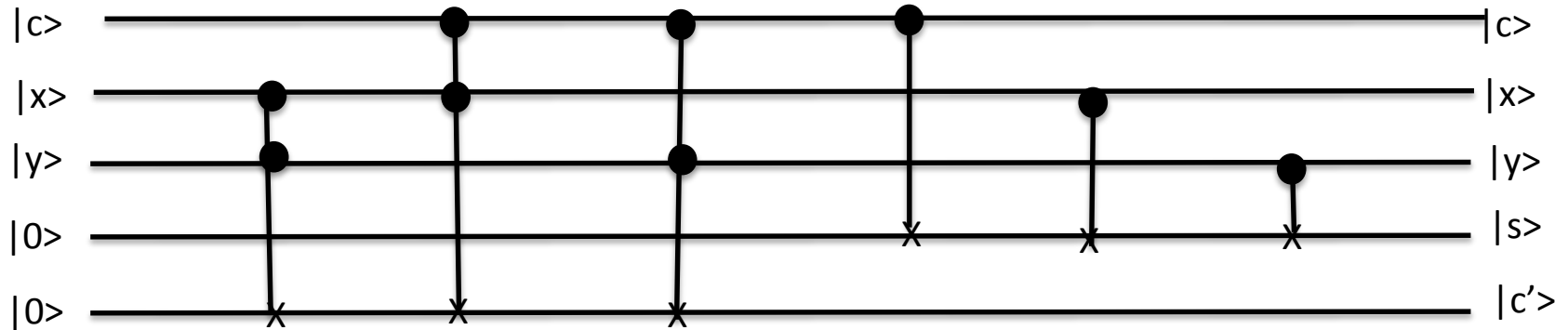
Numerous Types of Quantum Circuits

- Correct combinations of gates represented by single bit rotations and universal gates can function as
 - Reversible adders / subtractors
 - Half adders / subtractors
 - Full adders / subtractors
 - etc...

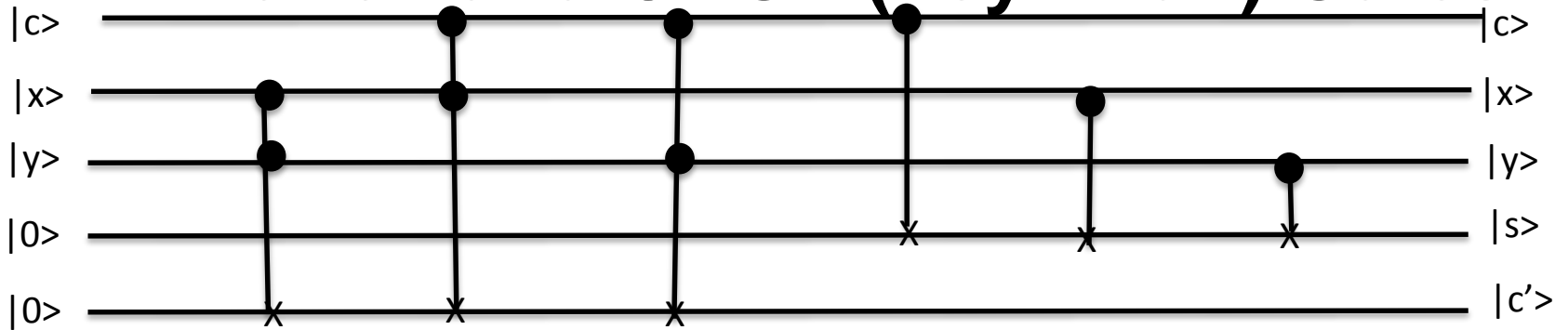
Example of a Quantum Circuit

- Construct a 1 bit full adder using Toffoli gates and controlled CNOT gates
- This quantum circuit has 5 inputs
 - $|x\rangle$ and $|y\rangle$ are the data bits
 - $|c\rangle$ is the incoming carry bit
 - $|s\rangle$ is the sum of $|x\rangle$ and $|y\rangle$ (modulo 2)
 - $|c'\rangle$ is the new carry bit

1 bit Full Adder Circuit with Toffoli and CNOT (Feynman) Gates

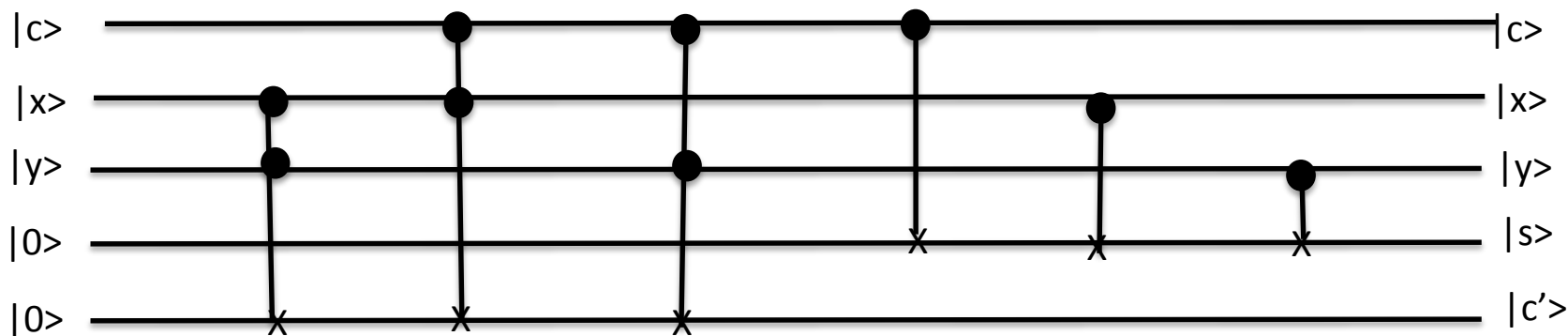


1 bit Full Adder Circuit with Toffoli and CNOT (Feynman) Gates



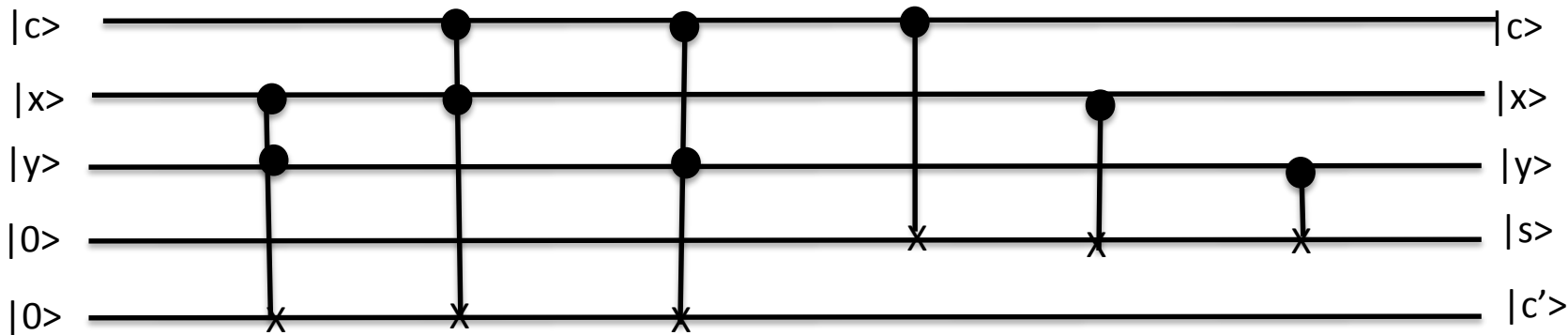
C	1						
X	0						
Y	1						
0	0						
0	0						

1 bit Full Adder Sample Calculation



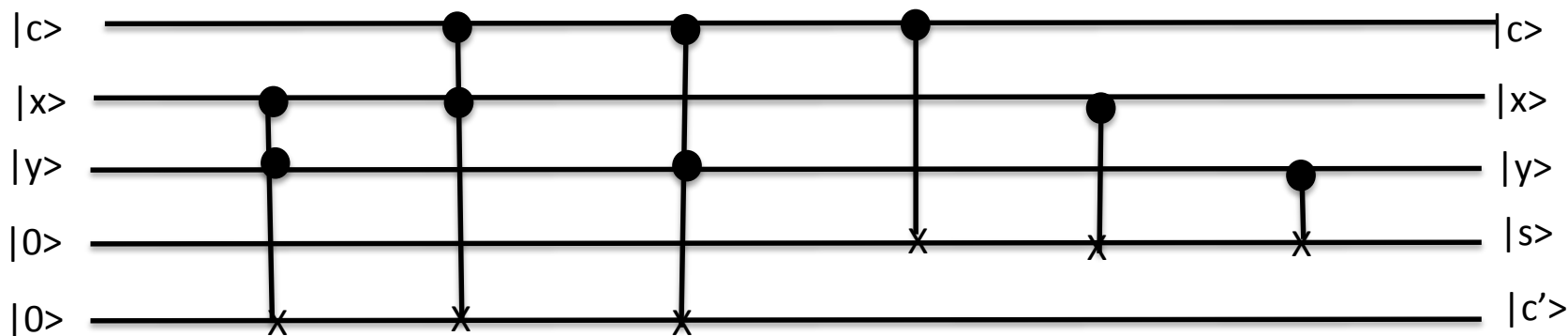
C	1	1					
X	0	0					
Y	1	1					
0	0	0					
0	0	0					

1 bit Full Adder Sample Calculation



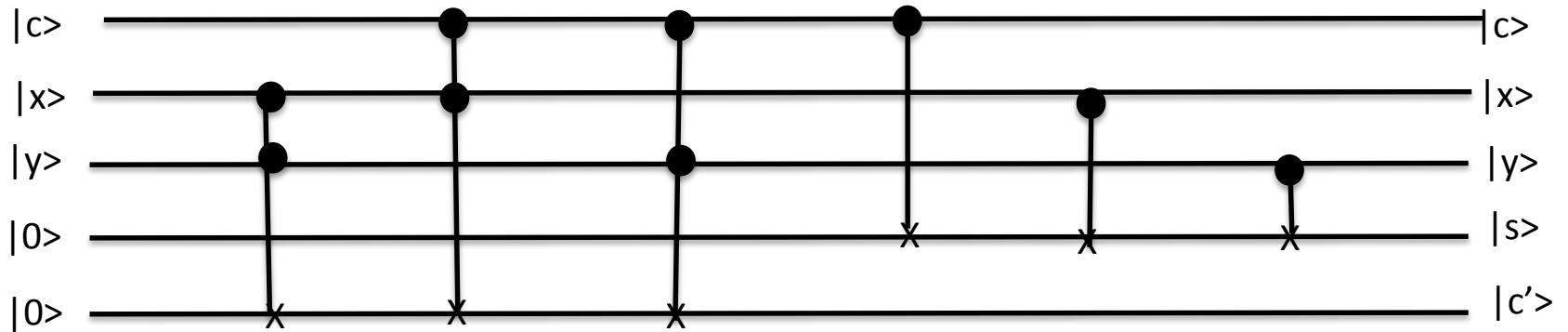
C	1	1	1				
X	0	0	0				
Y	1	1	1				
0	0	0	0				
0	0	0	0				

1 bit Full Adder Sample Calculation



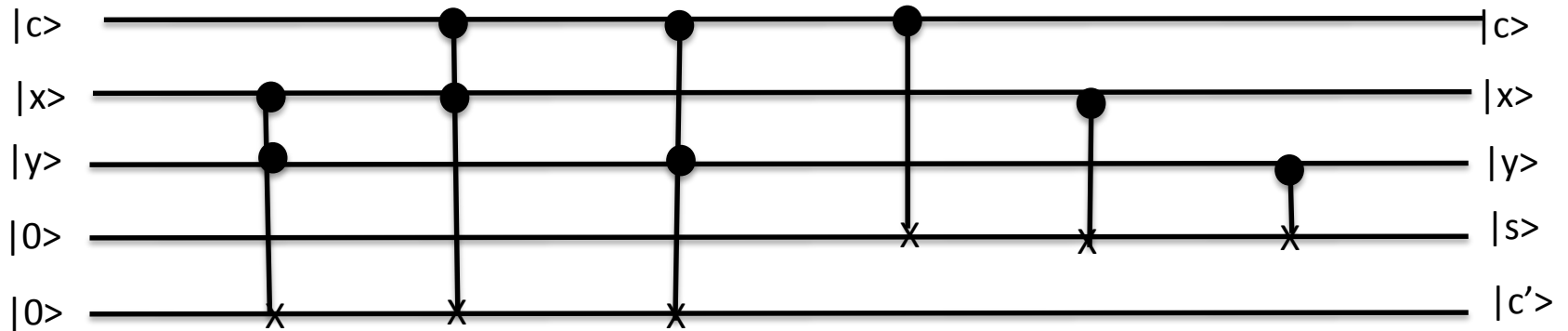
C	1	1	1	1			
X	0	0	0	0			
Y	1	1	1	1			
0	0	0	0	0			
0	0	0	0	1			

1 bit Full Adder Sample Calculation



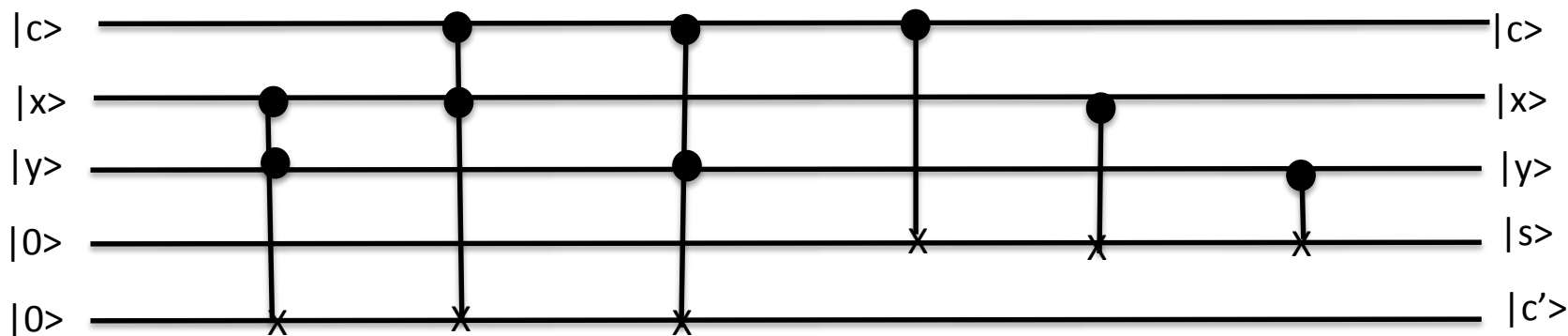
C	1	1	1	1	1		
X	0	0	0	0	0		
Y	1	1	1	1	1		
0	0	0	0	0	1		
0	0	0	0	1	1		

1 bit Full Adder Sample Calculation



C	1	1	1	1	1	1	
X	1	1	1	0	0	0	
Y	1	1	1	1	1	1	
0	0	0	0	0	1	1	
0	0	1	0	1	1	1	

1 bit Full Adder Sample Calculation



C	1	1	1	1	1	1	1	c>
X	0	0	0	0	0	0	0	x>
Y	1	1	1	1	1	1	1	y>
0	0	0	0	0	1	1	0	s>
0	0	0	0	1	1	1	1	c'>

Comparison of Classical and Quantum Aspects of Computation *

CLASSICAL vs. QUANTUM BITS	Cbits	Qbits
States of n Bits	$ x\rangle_n, 0 \leq x < 2^n$	$\sum \alpha_x x\rangle_n, \sum \alpha_x ^2 = 1$
Subsets of n Bits	Always have states	Generally have no states
Reversible operations on states	Permutations	Unitary transformations
Can state be learnt from Bits?	Yes	No
To get information from Bits	Just look	Measure
Information acquired	x	x with probability $ \alpha_x ^2$
State after information acquired	Same: still $ x\rangle$	Different: now $ x\rangle$

* arXiv:quant-ph/0207118v1 19 Jul 2002

Design Constraints for Building a Quantum Computer

Modifications Needed to Map From Classical to Quantum Computing

- A classical computer with “n bits” can have 2^n special orthonormal states (classical basis)
- A quantum computer may have arbitrary unit vectors from the entire vector space consisting of all linear combinations of classical basis states with complex coefficients (called amplitudes)
- Initial state will result in superposition of all corresponding output values
- (Note - superposition of all possible states is the origin of the expected exponential computational speedup)

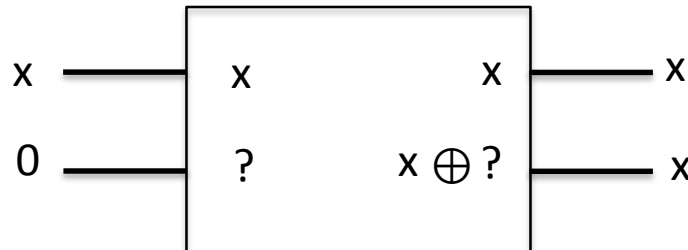
Modifications Needed to Map From Classical to Quantum Computing

Measurements

- Measurements in a classical computer are not a factor in the overall computational process
- This is not true for quantum computing
- From axiom 4 of quantum mechanics
a state evolves over time and is expressed mathematically by a unitary operator (transformation) for a closed quantum mechanics system
- This requires that a quantum gate must be reversible under unitary time evolution

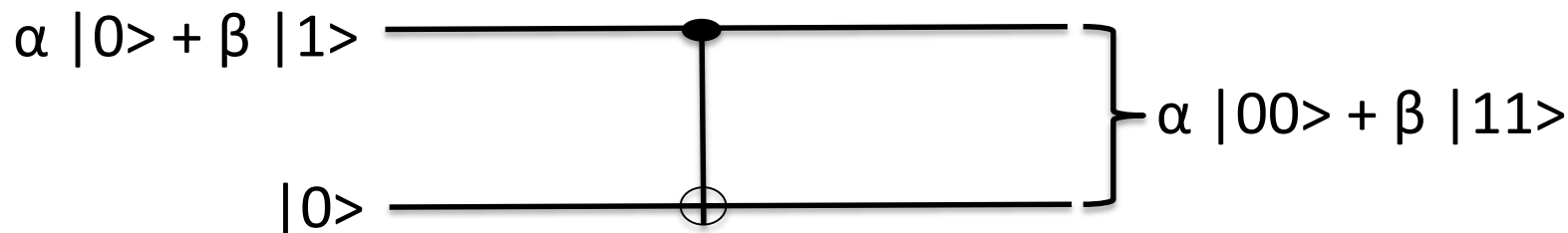
Quantum Information No Cloning Theorem

- A CNOT gate can copy a classical bit in some unknown state “x” and an additional bit initialized to zero and provide an output where both bits are in a state “x”



Quantum Information No Cloning Theorem

- Consider the CNOT quantum gate and a linear superposition state $\alpha |0\rangle + \beta |1\rangle$ and an additional bit initialized to zero



- Quantum mechanically this output is not possible because the general state vector $|a\rangle|a\rangle = (\alpha |0\rangle + \beta |1\rangle) (\alpha |0\rangle + \beta |1\rangle)$
 $|a\rangle|a\rangle = \alpha^2 |00\rangle + \alpha\beta |01\rangle + \beta\alpha |10\rangle + \beta^2 |11\rangle$
- In general $\alpha^2 \neq 0$ and $\beta^2 \neq 0$ and so the quantum circuit does not copy the part of the state vector with the terms $\alpha\beta |01\rangle + \beta\alpha |10\rangle$
- The no-cloning theorem states that it is impossible to create an identical copy of an arbitrary unknown quantum state
- This implies that signal fanout is not permitted

Measurements on a Quantum Computer

- Start with two quantum systems 1 and 2 that can interact with each other
- The act of measurement entangles the two systems quantum mechanically
- Entanglement destroys the superposition of states of system 1 so that some of the relative phases of the system 1 superposition are no longer present
- Result is a collapse of the states of system 1 that cannot be re-constructed

Coding Reversibility into a Quantum Circuit

- The action of every reversible quantum gate can be represented by matrix multiplication, where the matrix has the additional property of preserving the length of vectors. Such matrices are called “unitary” and are characterized by the equation $A^\dagger A = I$
- For gates represented by a matrix, the unitarity condition is necessary and sufficient for ensuring that pure states get mapped to pure states
- Because qubit states can be represented as points on a sphere, reversible one-qubit gates can be thought of as rotations of the Bloch sphere. This is why such quantum gates are often called “rotations”
- Quantum circuits are constructed from the combined actions of unitary transformations and single bit rotations

Now Have the Necessary Building Blocks to Construct a Quantum Computer

1. Understand concepts of a single qubit and two qubits
2. Have basic Linear Algebra to mathematically describe how qubits are expressed and transform over time
3. Have the basic postulates of quantum mechanics to verify that qubit transformations obey properties of QM
4. Identified and categorized all types of 1 qubit transformations (rotations)
5. Constructed several universal (reversible) gates that transform according to the postulates of QM
6. Introduced idea of quantum circuits that use rotation gates and universal gates to evolve qubits in time
7. Now can build a quantum computer and design/assemble sequences of rotation and universal transformation that can “program” this quantum computer

– Next Step –
**How does one build a quantum computer
that can execute a “program”
constructed of rotation and universal
gates?**

Last Slide