

The Quantum Computing Survival Guide

Written by Zack Schieberl

QCSG

Chapter 1: Why Quantum Computing?

- 1.1 History of Quantum Computers
- 1.2 The New Age of Computing
- 1.3 Modern Applications of Quantum Computers
- 1.4 Future Applications of Quantum Computers

Chapter 2: Qubits & Superposition

- 2.1 Quantum Particles
- 2.2 Modeling Waves with Vectors
- 2.3 Dirac Notation
- 2.4 Bloch Sphere Model
- 2.5 Transforming a Qubit
- 2.6 Sequential Transformations
- 2.7 Beam Splitter
- 2.8 Qubit Implementations

Chapter 3: Quantum Circuits

- 3.1 Definition of a Gate
- 3.2 Pauli Gates
- 3.3 Named Gates
- 3.4 Quantum Registers
- 3.5 CNOT Gate
- 3.6 Controlled-U Gate
- 3.7 Reversible Computing

Chapter 4: Entanglement

- 4.1 The Tensor Product
- 4.2 Superposition of Quantum Registers
- 4.3 Generating Entanglement
- 4.4 Bell States

Chapter 5: Quantum Networks

- 5.1 No-Cloning Theorem
- 5.2 Teleportation
- 5.3 Superdense Coding

Chapter 6: Quantum Cryptography

Chapter 7: Quantum Error Correction

Chapter 8: Quantum Algorithms

Chapter 9: Quantum Computer Architecture

Chapter 10: Quantum Compilers

Chapter 11: Simulating Quantum Computers

Chapter 12: Quantum Artificial Intelligence

Chapter 13: Chemical Simulation

Chapter 1: Why Quantum Computing?

This entire document is used to describe what quantum computing is, but I think it is important to understand why quantum computers are important. What is their history and what can they do for society?

1.1 History of Quantum Computers

idk what it is lol

1.2 The New Age of Computing

Comparing classical computers to quantum. Moore's Law

1.3 Modern Applications of Quantum Computers

cool stuff

1.4 Future Applications of Quantum Computers

super cool stuff

Chapter 2: Qubits & Superposition

The smallest possible unit of data in a classical computer is a *bit*. It stands for binary digit. A single bit can either be zero or one. The term 'bit' is often used interchangeably with *boolean*, whose value is either true or false. Every piece of data which can be stored in a computer's memory is a sequence of these bits. Qubits are the smallest possible unit of data in a quantum computer and they have more properties than ordinary bits.

2.1 Quantum Particles

Although this document is focused towards computer scientists, it is important to learn about the nature of quantum particles to understand where qubits come from. This explanation will only scrape the surface of the wonderful world of quantum particles, but it will suffice for the purposes of this document.

Before discussing quantum particles, let's talk waves. A *wave* is any mathematical function which continues to repeat itself, forever. Think about the waves at the beach, how they go up and down as far as you can see. The distance until the wave starts to repeat itself is called the wave's *period*.

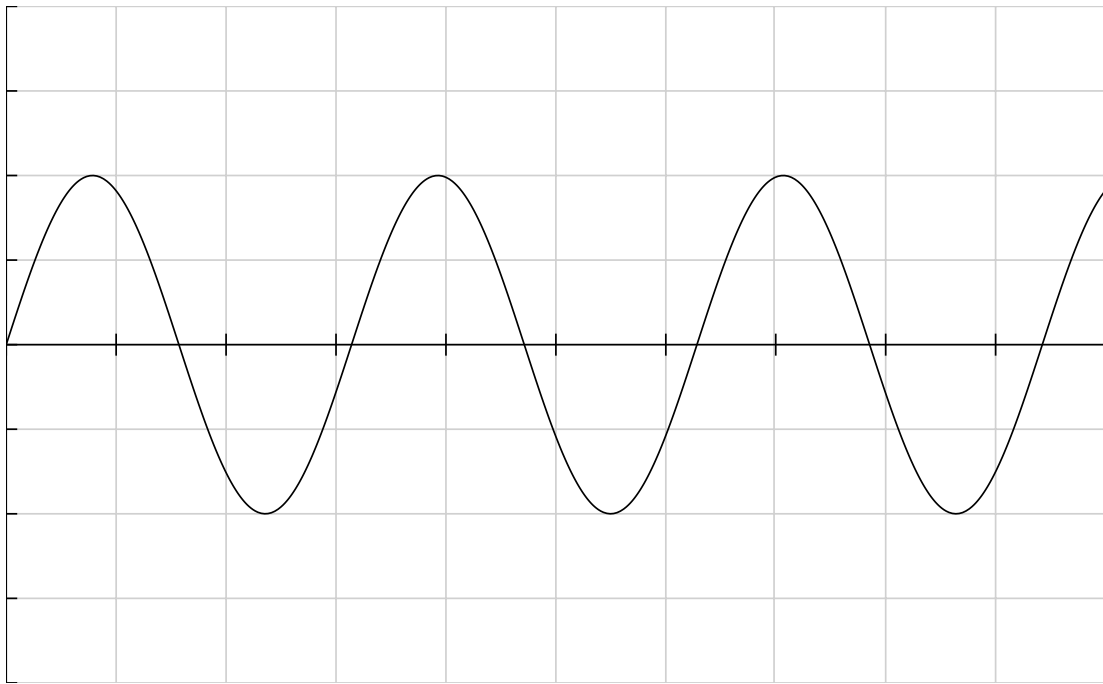


Figure 2.1.1 The Sine Wave

Several famous experiments, such as the double slit experiment, have shown that light and matter can have properties of particles and waves. This is known as *wave/particle duality*. When an electron is released from a source, it acts as a wave which propagates through space until contacting a surface. Once the electron collides with a surface, it is only observable in one location.

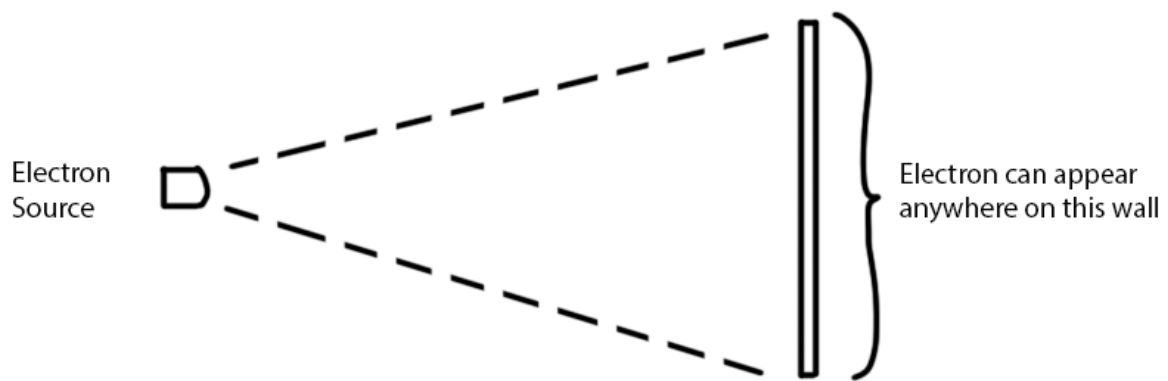


Figure 2.1.2 - Electron Colliding with a Wall in a Random Location

The location where it will be observable is random and cannot be determined ahead of time. However, the likelihood of the electron appearing in a specific location can be calculated using a probability function.

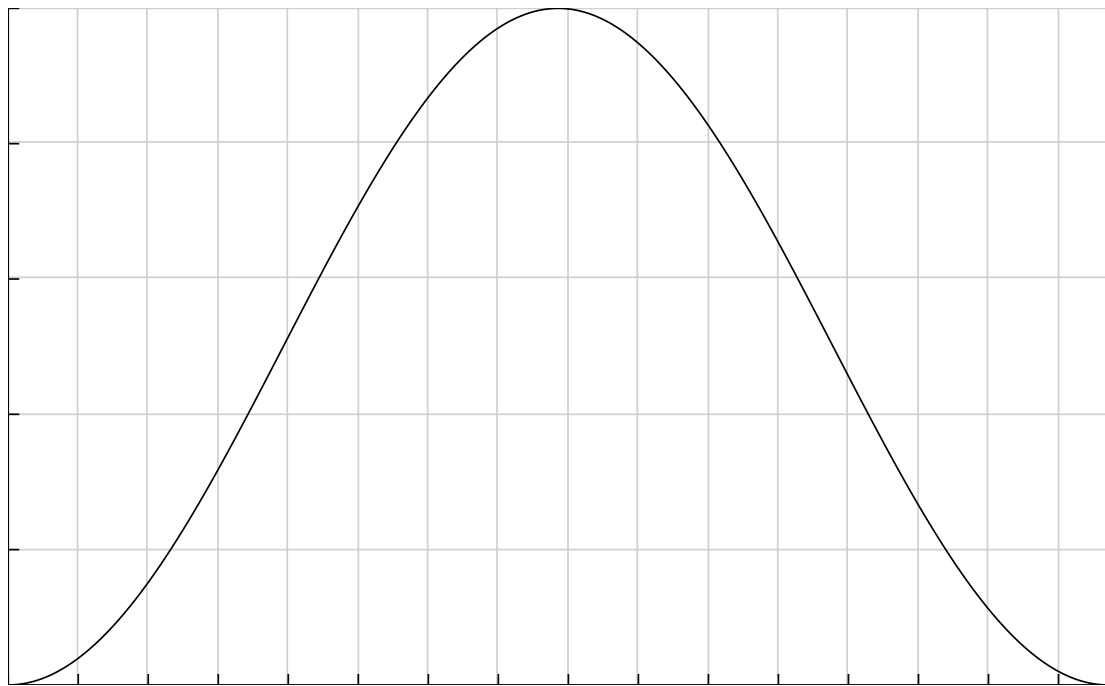


Figure 2.1.3 - A Probability Function

This function shown above contains information on how likely the electron is to be observed at a specified location. In this graph, the X axis is the location and the Y axis is how likely it is to appear at that location. The electron is very likely to appear somewhere in the middle and it is very unlikely to appear near the edges. In the bottom left corner of this graph, when $x = 0$ it shows that $y = 0$ as well. This means the electron can **never** appear at location 0. The same logic applies at the bottom right corner.

The wave equation is a model for how waves move through space. It is a function of two variables, location and time: $F(x, t)$. At the moment the electron is observed, time is no longer a variable and is held constant since it does not change. In quantum computing, how the wave moves is not a concern of ours, so we do not use time as a variable.

This section is self contained, the information presented here isn't used in the following chapters. It is included to provide a foundation for where qubits come from. The rest of this document is dedicated towards explaining the rules governing qubits and their interactions, this is the only section which makes an attempt at showing where the rules are derived from. An inquiring mind may ask "why do qubits follow these rules specifically?" and the only answer is because they follow the laws of quantum mechanics.

2.2 Modeling Waves with Vectors

The fundamental unit of information in classical computers are bits. They can either be on or off. There are only two possible states. Figure 2.1.2 shows a probability function which is continuous, there are an infinite number of locations, each with their own probability of being observed at said location. To make these probability functions palatable to computer scientists, the probability functions need to be transformed into binary.

There is no particular reason why we must make the continuous function binary, we could divide it into 3 sections and transformed into a *trit*, the fundamental unit of information with three states. We could also divide it into 4, 5, 10, 29, 7326, or any number of discrete sections. We could even leave it continuous, without ever dividing it into sections. The only reason for dividing the function into two sections is because there are already many decades worth of development in computer science which is in binary. Having two states allows quantum computer scientists to create algorithms, build architectures, and store data based on classical computers without factoring in a change of radix.

To extend our example from the previous section, we can convert to binary by drawing a line across our wall to see if the electron is observed above or below the line. The electron has a probability α^2 of being above the line where α^2 is $0 \leq \alpha^2 \leq 1$. There is also a probability of $\beta^2 = 1 - \alpha^2$ that the electron is below the line where β^2 is also $0 \leq \beta^2 \leq 1$.

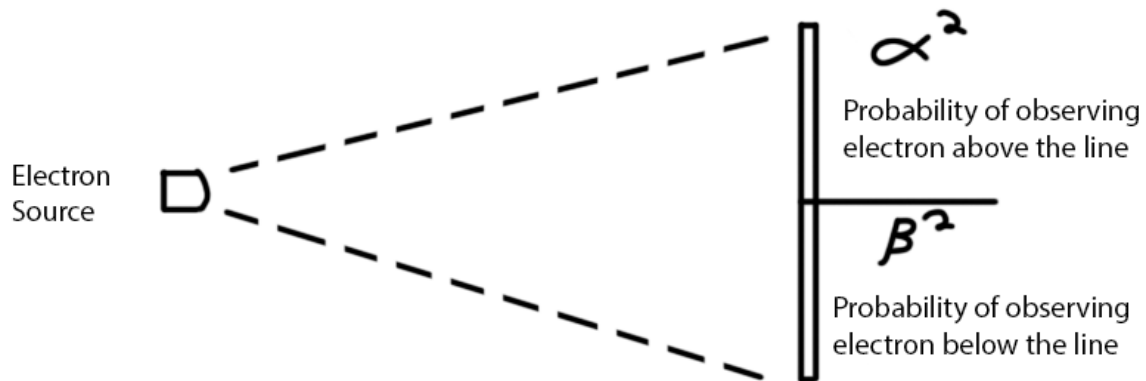


Figure 2.2.1 - Electron Colliding with a Discretely Sectioned Wall

What we have just described is known as a *probabilistic bit*, or *pbit*. A pbit is a bit which has a certain probability of being off, which we have denoted above as α^2 . It also has a certain probability of being on, which we have denoted above as β^2 . As both these values are probabilities, they are constrained to $\alpha^2 + \beta^2 = 1$. The concept of a pbit is not exclusive to quantum, they also are used in binary classifiers from machine learning, random number generators, etc.

Our goal is to convert a wave equation like the sine function into a pbit. One of the first things a precalculus course teaches is the relationship between trigonometric functions and circles. Taking a look at the unit circle, we can see the X and Y coordinates of the circle with radius 1 are the outputs of the cosine and sine functions.

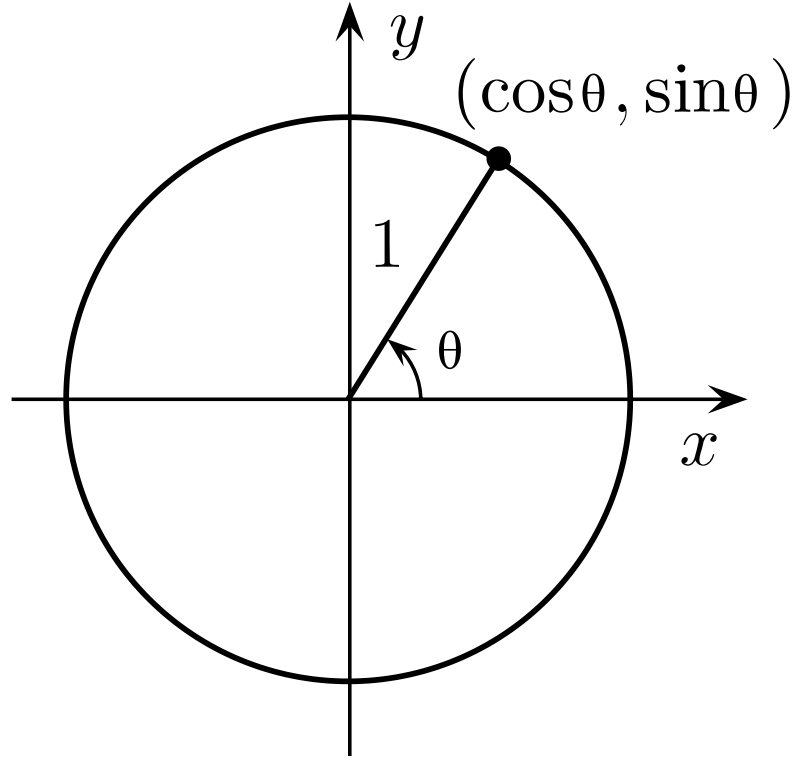


Figure 2.2.2 - The Unit Circle

A *vector* is a mathematical object which has both magnitude and direction. We have a vector here whose magnitude is 1 and direction is determined by the variable θ , the angle from the positive X axis to the vector. θ is called a *polar coordinate* which is the input to the trigonometric functions that produce the cartesian coordinates x and y . The vector \mathbf{v} is described as:

$$\mathbf{v} = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix}$$

This can be used to fit our description of a pbit by remembering the Pythagorean trigonometric identity:

$$\begin{aligned} x^2 + y^2 &= 1 \\ \cos^2 \theta + \sin^2 \theta &= 1 \\ \alpha^2 + \beta^2 &= 1 \end{aligned}$$

From this we can see our probabilistic bit relate to the quantum wave functions as:

$$\begin{aligned} \alpha^2 &= \cos^2 \theta \\ \beta^2 &= \sin^2 \theta \end{aligned}$$

Since α^2 and β^2 are always positive, we can apply the constraint $0 \leq \theta \leq \frac{\pi}{2}$.

There are two unit vectors which are special enough to be named: \hat{i} and \hat{j} . These vectors are both of length 1 and are orthogonal to each other.

$$\begin{aligned} \hat{i} &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ \hat{j} &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{aligned}$$

These *basis vectors* represent our X and Y axes since \hat{i} is directly along the X axis and \hat{j} is directly along the Y axis. The vector \mathbf{v} can be rewritten using vector addition:

$$\mathbf{v} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \alpha \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \beta \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \alpha \hat{i} + \beta \hat{j}$$

After viewing this relationship, we see:

- \hat{i} conceptually represents the off state, or 0, or the electron being observed above the line on the wall
- \hat{j} conceptually represents the on state, or 1, or the electron being observed below the line on the wall
- α^2 and β^2 still represent the probabilities of those states occurring
- \mathbf{v} represents the entire state of the pbit, or the electron & wall system

This model of a pbit is not a completely new innovation, it is simply a transformation from the quantum wave equation which is easier to digest for computer scientists.

However, our transformation is not complete. We have translated the probabilistic nature of quantum particles but there is still another feature of those particles which we have not translated: phase. Without including phase, we cannot say our model acts as a true quantum bit, also known as qubit.

Before extending our vector based model to the true qubit model, a new style of notation needs to be introduced. The notation of quantum physicists: the Dirac notation.

2.3 Dirac Notation

Paul Dirac invented a notation system for discussing quantum mechanics and it has been adopted into quantum computing. The notation is called "bra-ket" since it uses the angle brackets $\langle \rangle$. Truly the peak of twentieth century physics humor.

Ket Notation

Any vector can be expressed by placing it inside a vertical line ($|$) and right angle bracket (\rangle). By placing it inside, $|\mathbf{v}\rangle$, we pronounce it "ket-v". This object conceptually represents a quantum state, however it is still mathematically equivalent to a vector. Specifically, a quantum state is a member of a *complex Hilbert space*, but thinking of them as vectors is acceptable in quantum computing.

Since a ket represents a quantum state, we can put our own quantum states inside the ket like: $|\text{above-line}\rangle$ and $|\text{below-line}\rangle$ to represent the electron being observed above or below the line on the wall. Some more examples of kets are:

- $|\text{spin-up}\rangle$ and $|\text{spin-down}\rangle$ to represent the spin of a photon
- $|\text{ground}\rangle$ and $|\text{excited}\rangle$ to represent how close an electron is to its nucleus
- $|/\rangle$ and $|\backslash\rangle$ to represent diagonal and anti-diagonal polarization of light
- $|\text{clockwise}\rangle$ and $|\text{counterclockwise}\rangle$ to represent direction of current flow in a circuit

However by far the most common kets used in this document and in the quantum computing community are $|0\rangle$ and $|1\rangle$. We can abstract away the physics used for the quantum state and focus on the value. Just like how computer scientists don't care if their bit is representing a magnetic field in their hard drive or amplitude of current in a wire, they just care about the value. $|0\rangle$ and $|1\rangle$ can be any two orthogonal basis states in quantum mechanics.

The quantum state representing an entire system is denoted by the special Greek letter ψ , written "Psi", and pronounced "Sai". Using all of this information, we can rewrite our pbit model using Dirac notation:

$$\begin{array}{ll} \text{Vector Notation} & \mathbf{v} = \alpha \hat{i} + \beta \hat{j} \\ \text{Dirac Notation} & |\psi\rangle = \alpha|0\rangle + \beta|1\rangle \end{array}$$

Even though we have not extended the pbit to the full qubit model, the above equation still holds for qubits. This equation is called **The General Form of the Qubit** and is one of the most important concepts in all of quantum computing.

Using our new form of notation, let's reason about the process of projecting an electron at a wall. If the electron is detected above the line, then we know $|\psi\rangle = |0\rangle$. This is the same as saying "the state of the electron is the state of being above the line". On the other hand, if the electron is detected below the line, we know $|\psi\rangle = |1\rangle$. This is the same as saying "the state of the electron is the state of being below the line".

However, what do we call the state before the electron contacts the wall? The electron is still a wave moving through space. We do not know if the electron will be observed above or below the line since they are random, but we know the likelihood of those events occurring $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. Until the electron makes contact with the wall, it is reasonable to say "the state of the electron is both the state of being above and below the line". In other words, the electron's state is a combination of the two outcome states. When a quantum state can only be described as a combination of two or more basis states, we say the quantum state is in *superposition*.

Superposition

Superposition is one of those words which people think is more magical than it really is. Flipping a coin can be described with superposition too. When the coin is in the air, before it lands, the state can be modeled as: $|\text{coin}\rangle = \frac{1}{\sqrt{2}}|\text{heads}\rangle + \frac{1}{\sqrt{2}}|\text{tails}\rangle$. When this coin is twirling around in the air, you have no idea which side it will land on. At this moment in time, it is reasonable to say the coin is a combination of heads and tails. The coin is in superposition. To a human, the outcome of a coin flip is a random event. The only difference between the coin and the electron is if you have a good enough computer, it could simulate the physics of the coin toss and determine the outcome before the coin landed whereas the outcome of the electron cannot be determined before it lands.

When the electron collides with the wall, it is observed. The moment the electron is observed, the state goes from being in superposition to a basis state. This process is called *collapsing* and is instantaneous. The probabilities α^2 and β^2 are destroyed and only the state $|0\rangle$ or $|1\rangle$ is left. Colliding with a wall is not the only way to collapse a quantum state. Making any sort of measurement or observation on the system will cause the superposition to be destroyed and the system will choose one of the basis states to collapse to using the assigned probabilities.

Erwin Schrödinger is a famous physicist who was instrumental in developing the quantum wave equation. He also created a hypothetical scenario to help explain the concept of superposition: *Schrödinger's Cat*. This thought experiment describes putting a cat inside a soundproof box with a bomb which has a 50% chance to explode after 1 minute passes. After a minute passes, there are two possible outcomes of the experiment:

- The bomb explodes, and the cat is dead
- The bomb does not explode, and the cat is alive

However, to see if the cat is alive or not, the box must be opened. Let's begin the experiment. Put the cat inside the box, close it, and wait a minute. After the minute passes, what is the state of the cat? Is it alive or dead? We don't know. Nobody does, until the box is opened and the state is observed. Before the box is opened, we can say the state of the cat is in superposition. The cat is a combination of dead and alive at the same time. We can also model the state of the cat as: $|\text{cat}\rangle = \frac{1}{\sqrt{2}}|\text{alive}\rangle + \frac{1}{\sqrt{2}}|\text{dead}\rangle$.

It is important to note that no cats were harmed in this theoretical scenario.

Bra Notation

row vectors, conjugate-transpose, overlap = scalar (called linear functional/operator, calculating probability amplitudes w/ overlap, $\langle \psi | \psi \rangle = 1$, inner/outer products

2.4 Bloch Sphere Model

Adjusting the Pbit

Now that we have a proper system of notation to discuss quantum states, we will extend our model of a pbit to the full model of a quantum bit, also called a *qubit*. Quantum particles have the ability to randomly collapse to a basis state, but they also have another key attribute called *phase*. Finding where phase comes from can be a tricky subject to cover without diving headfirst into quantum mechanics. For the sake of brevity, we will not be covering how phase impacts the quantum wave equation, only how phase impacts the qubit. We need to transform the pbit model to accommodate phase.

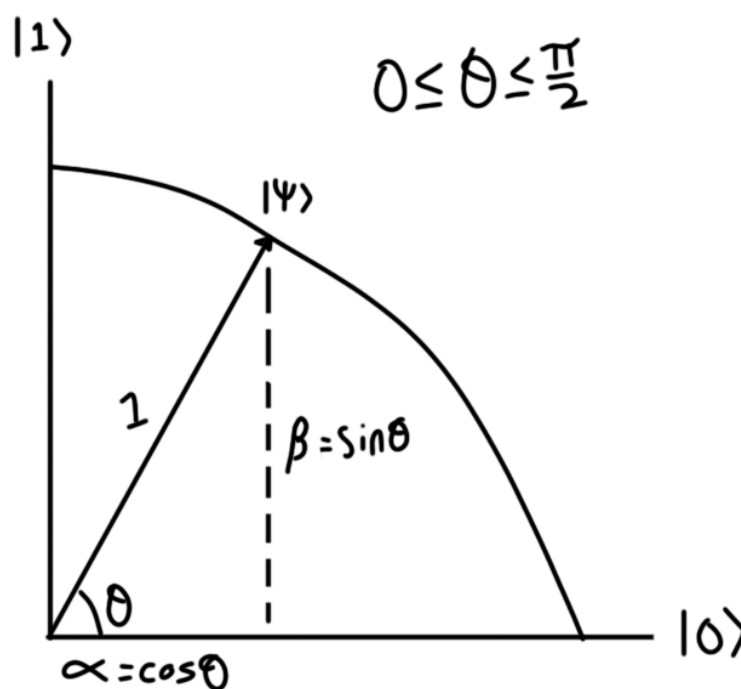


Figure 2.3.1 - The Pbit Model

This is the pbit model we found from dividing the wave equation into two discrete sections using the unit circle. There are two transformations we need to make to our pbit to have it match the physicist's model of a qubit.

- The first is to switch the $|0\rangle$ and $|1\rangle$ axes. This is equivalent to reflecting the model over the line created by $y = x$. Now $|0\rangle$ represents the vertical axis and $|1\rangle$ represents the horizontal axis. Notice θ is still the angle between $|0\rangle$ and $|\psi\rangle$.
- The second transformation is to pull $|1\rangle$ down so it is also vertical. This makes both $|0\rangle$ and $|1\rangle$ vertical and changes the definition of θ , α , and β .

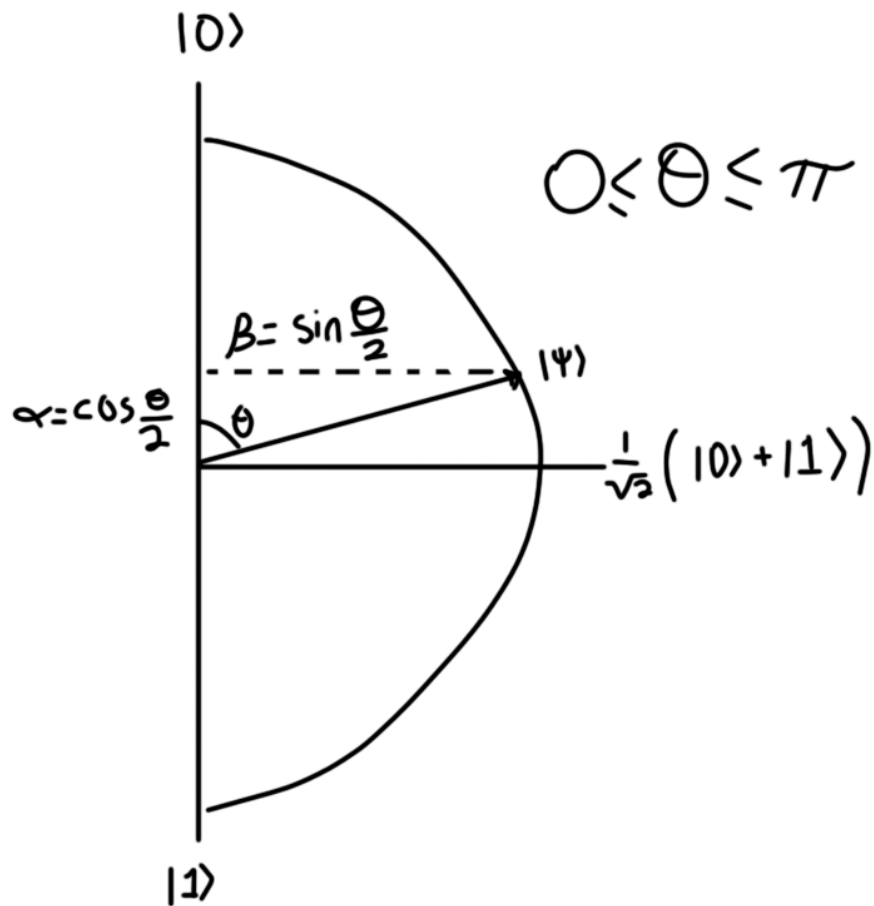


Figure 2.3.2 - The Partial Qubit Model

The above model has some interesting things to note about it. We have changed the definition of θ so it is bounded above by π instead of $\frac{\pi}{2}$. This has also changed how the basis coefficients are defined. Now $\alpha = \cos(\frac{\theta}{2})$ and $\beta = \sin(\frac{\theta}{2})$. The basis states $|0\rangle$ and $|1\rangle$ are mathematically still orthogonal, even if it doesn't appear that way visually. The horizontal state is now in *equal superposition*, which occurs when the probability of all basis states are the same.

$$\begin{aligned}
 |\psi\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \\
 \left(\frac{1}{\sqrt{2}}\right)^2 + \left(\frac{1}{\sqrt{2}}\right)^2 &= 1 \\
 \frac{1}{2} + \frac{1}{2} &= 1
 \end{aligned}$$

This partial model of the qubit is now ready to introduce phase.

Spherical Coordinates

Polar coordinates are a system of coordinates in 2 dimensions which use a distance from the origin and an angle to describe every possible point: (r, θ) . Our pbit model used polar coordinates but restricted r to always be equal to 1. To introduce phase, we need to add another variable to our model and add another dimension.

Spherical coordinates are a system of coordinates in 3 dimensions which use a distance from the origin, an angle from the positive Z axis, and an angle from the positive X axis to describe every possible point: (r, θ, ϕ) . In this model, θ is the angle from the positive Z axis and is called the *inclination angle*. ϕ is the angle from the positive X axis and is called the *azimuth angle*.

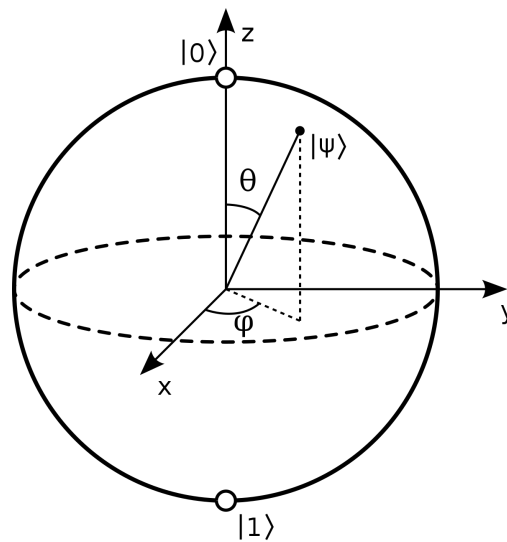


Figure 2.3.3 - The Bloch Sphere

The **Bloch Sphere**, named after Felix Bloch, is the full model of the qubit. It uses spherical coordinates to describe every possible state a quantum particle can occupy. A *pure state* is any quantum state which lies on the surface of the unit sphere with radius 1. A quantum state could also be a *mixed state* which has radius less than 1, but will not be discussed here. Here are some facts about our new Bloch Sphere model:

- The $|0\rangle$ and $|1\rangle$ states are still vertical and orthogonal to each other.
- The inclination angle, θ , is still bounded as $0 \leq \theta \leq \pi$
- The azimuth angle, ϕ , has the bounds: $0 \leq \phi < 2\pi$

What about our basis coefficients? Did α and β change?

Let's Talk Phase

The definitions of α and β did change, but before we talk about that, what is up with phase? We were forced to bring our pbit model into another dimension so we could incorporate phase to make a qubit. How does that help? Let's take a look at the probability constraints from before:

$$\begin{aligned} 0 &\leq \alpha^2 \leq 1 \\ 0 &\leq \beta^2 \leq 1 \\ \alpha^2 + \beta^2 &= 1 \end{aligned}$$

These constraints are still true with the Bloch Sphere, even if the definitions of α and β changed. Here's the thing though: every possible value of α^2 and β^2 were accessible already in the pbit model. If we think of α^2 as a function of θ : $\alpha^2(\theta) = \cos^2(\frac{\theta}{2})$, we see it is a 1-1 function since $0 \leq \theta \leq \pi$!

This realization only makes phase even more confusing! Why would we go through all the effort of adding another dimension if we could already access every value?

It is true that α^2 is always positive so any value can be accessed by the correct θ value, but what about α ? There is no guarantee α needs to be positive, it could be negative

global phase vs relative phase

2.5 Transforming a Qubit

Rotation around axis -> new angles

new angles -> rotations around axis(es)

state space, accessible state space, complete state space

2.6 Sequential Transformations

Any transformation from one coord to another can be accomplished in 2 rotations

You only need 2 axes to rotate to any position: XZ, YZ, or XY

You can also swap the order XZ -> ZX by editing the rotations

2.7 Beam Splitter

2.8 Qubit Implementations

superconductors

ion trap

photon

more about this in quantum architecture

Chapter 3: Quantum Circuits

yum

3.1 Definition of a Gate

boolean function

classical gates

matrix

reversible

universal set of gates

3.2 Pauli Gates

single qubit rotations

their matrices

half a bit flip (both x and y)

3.3 Named Gates

Hadamard, S, T

Hadamard can either be ZY.5 or Y.5X

3.4 Quantum Registers

3.5 CNOT Gate

Interact with multiple qubits

Swap = 3 cnots

3.6 Controlled-U Gate

Any controlled gate!

3.7 Reversible Computing

Chapter 4: Entanglement

swoosh

4.1 The Tensor Product

distributing vectors and distributing kets

4.2 Superposition of Quantum Registers

4.3 Generating Entanglement

mixed states, $r < 1$

undistributable

EPR pair

information instantly

4.4 Bell States

hadamard \rightarrow CNOT

4 bell states

read 1 then know the other instantly

Chapter 5: Quantum Networks

it cool

5.1 No-Cloning Theorem

Heisenberg uncertainty principal

5.2 Teleportation

not like the movies

moving a qubit state over a long distance

distributing quantum pairs ahead of time

decoherence

5.3 Superdense Coding

teleportation in reverse

2 bits for the price of 1

Chapter 6: Quantum Cryptography

Chapter 7: Quantum Error Correction

Chapter 8: Quantum Algorithms

QFT, Shors, Grovers, Deutsch's Problem, Simon's Problem,

Chapter 9: Quantum Computer Architecture

Chapter 10: Quantum Compilers

phase normal form

#of cnots in circuit $> (n-1)!$

Chapter 11: Simulating Quantum Computers

classically!

matrix form

algorithm form: bitwise CNOT & bitwise X&Y&Z

cubeOffset instead of cubeDepth

phase $\neq 0$ instead of $== 1$

qiskit universal gate form

deferred form partitioning

Chapter 12: Quantum Artificial Intelligence

Chapter 13: Chemical Simulation
