PHYC90045 Introduction to Quantum Computing

Lab Session 1

1.1 Introduction

Welcome to Lab-1 of PHYC90045 Introduction to Quantum Computing.

The purpose of this first lab session is to:

- learn to use the Quantum User Interface (QUI) to program/simulate quantum circuits
- use the QUI to analyse single qubit states/gates and their representation on the Bloch sphere
- apply your knowledge to an application in quantum communication

1.2 The Quantum User Interface

The QUI is a web-based graphical user interface developed by the Hollenberg group at the University of Melbourne to program, simulate and analyse quantum circuits. The QUI allows the users to specify qubit number, build and simulate quantum circuits by easy drag-and-drop placement of quantum gates, and examine the quantum state at every time step in the circuit/program. The latter feature is critical to understanding QC, and distinguishes the QUI from other on-line QC programing/simulation tools.

1.3 Sign up and start the QUI

The QUI is accessed through a web-based interface, which at present is restricted to UoM staff and students. Sign up will occur at the start of the Lab-1 class.

Step 1: Open a web browser (preferably Google Chrome or Firefox), and go to the supplied URL.

Step 2: You will need to create an account to access QUI for the first time.

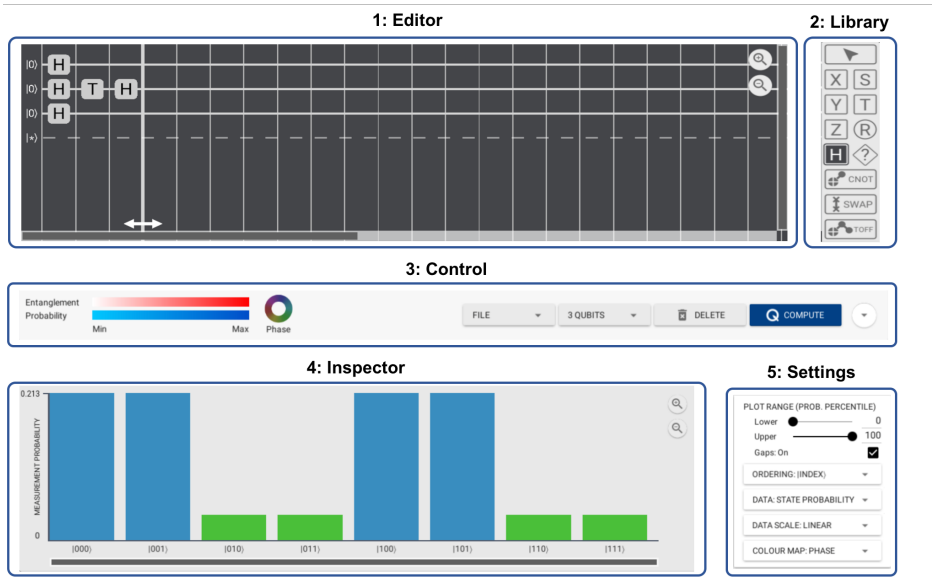
Note: **you must use your University of Melbourne email address as your login name.**

Follow the steps to create your account (including answering a few simple questions about use and education level).

Step 3: Once you have signed-up, start the QUI.

1.4 The QUI panels

Shown below are the various QUI panels.



Editor (Panel 1): This is where a quantum circuit is programed by dragging and dropping from the gate Library (Panel 2). On the left side, the initial states of three qubits (for this example) are shown in the default |0⟩ state. You can add more qubits using the |*⟩ icon (or QUBITS in the Control panel). The vertical lines separate time steps left to right. The thicker vertical with the left-right arrow is the "time-scrubber", which can be moved to analyse the quantum state at different times in the quantum circuit. The time-scrubber will also display the degree of entanglement at various parts of the quantum circuit (Lab 2).

Library (Panel 2): This panel shows the gate library (quantum operations), including measurement. Hovering the cursor over any single-qubit gate will display the animation of the corresponding operation on the Bloch sphere. Select one of these gates and place it in the Editor panel at the desired location(s). There is even a paint feature.

Control (Panel 3): The colour scales on the left side indicate the min-max range of various output parameters (probability, entanglement, phase). Next is a drop-down FILE menu (New, Save, Load, Save As). QUBITS allows the users to select the number of qubits. DELETE removes selected quantum gates from the circuit. COMPUTE will run the current circuit and display the results in the Inspector panel (more accurately, the circuit is sent to our quantum simulator and the entire set of results (all time points) is sent back to your QUI session for analysing – please don't sit there and hit compute a million times).

Inspector (Panel 4): Here you can visualise the quantum state (after compute) at any time point corresponding to the position of the sliding time-scrubber. By default, the horizontal axis is ordered according to the state index in the computational basis (|11⟩ = |3⟩ etc), and the vertical axis by default gives the probability of each state. Detailed information about the state amplitudes (magnitude and phase) is given in the State Info Card that pops up when the mouse hovers over a state in the histogram. Note the hazard warning symbol bottom right – this indicates the current Inspector plot does not contain up-to-date data (i.e. with respect to changes in the circuit shown in the Editor).

Settings (Panel 5): Here the user has additional control over the various plotting options in the Inspector, such as PLOT RANGE, ORDERING, and AXIS (data) options.

State Info Card (Panel 6): This panel appears when the cursor is on a particular state in the Inspector panel and will display all information about the state's identity (binary and integer index), complex valued amplitude (magnitude and phase), and probability.

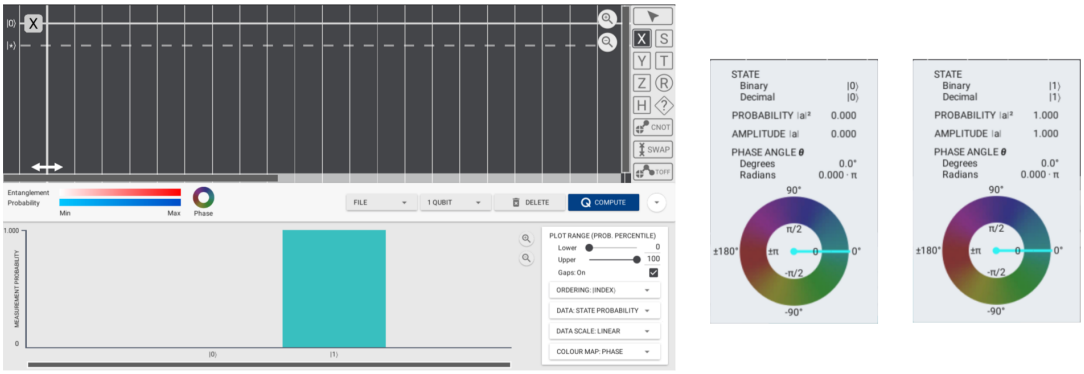
1.5 Quantum Gates, Amplitude, Probability and Phase

Now that you have started the QUI, the next step is to program some simple quantum gates to see how it works.

Let's start with a simple X gate. Recall that an X gate flips the state of a qubit: for example: $X|0\rangle \rightarrow |1\rangle$. To program a X gate in the QUI do the following:

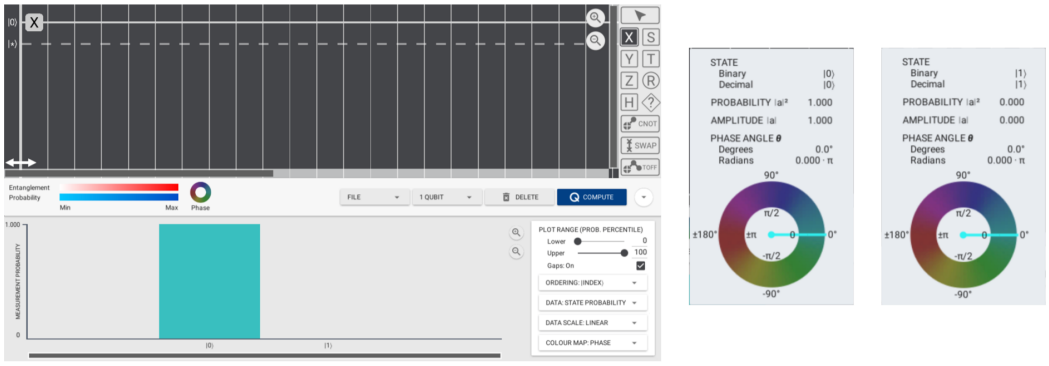
- select 1 QUBIT (Control Panel)
- click on an X gate (Library), place it in the first position on the qubit line (Editor).
- hit the COMPUTE button.

The results will look like below left:



The output state (time-scrubber after X) contains the $|1\rangle$ state with probability 1.0 (and zero phase) and the $|0\rangle$ state with zero probability (i.e. doesn't exist). Point to the relevant part of the plot to bring up the State Info Cards (SICs) – shown on right above.

Now drag the time-scrubber to the first time step to inspect the state before the application of the X gate. The results will look like below left, with the SICs for each state shown on the right:



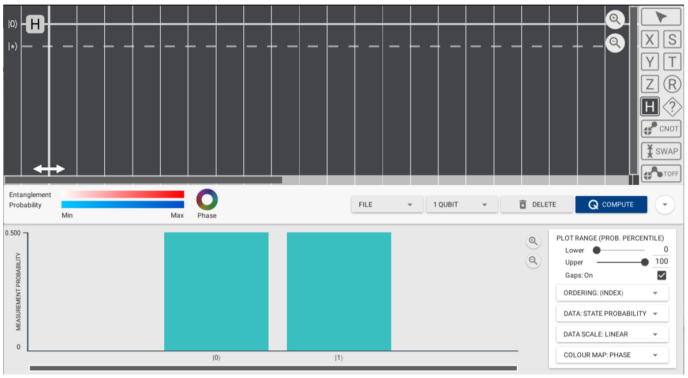
Before you go on further make sure you understand the information in the State Info Cards (ask a tutor, or refer to lecture notes).

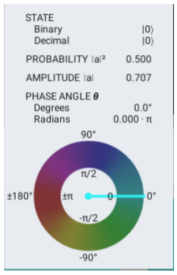
Now let's instead use a Hadamard gate (denoted by H). Recall that this gate places the qubit into an equal superposition state:

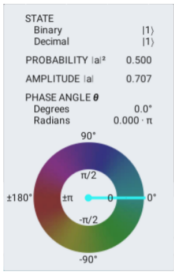
$$H|0\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

In order to test this gate in QUI:

- delete the X gate (using the DELETE button)
- replace it by a H gate from the Library.
- click on COMPUTE to find the outcome state after H operation:

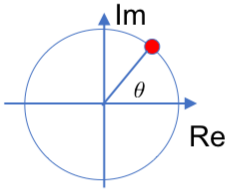






Quantitative information on the SICs indicates that both $|0\rangle$ and $|1\rangle$ states have equal amplitudes of $\frac{1}{\sqrt{2}}$, zero phase, and probabilities of $\left[1/\sqrt{2}\right]^2 = 1/2$.

In the above cases the phase angle of the amplitude was zero, but we will soon encounter examples where this is not the case. You may need to recall the basic transformations between Cartesian and polar descriptions of complex numbers:



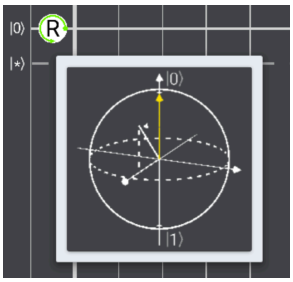
$$a = \mathrm{Re}[a] + i\,\mathrm{Im}[a] = |a|e^{i\theta} \rightarrow \begin{matrix} |a| = \sqrt{\mathrm{Re}[a]^2 + \mathrm{Im}[a]^2} \\ \theta = \tan^{-1}\left(\mathrm{Im}[a]/\mathrm{Re}[a]\right) \end{matrix}$$

$$e^{i\theta} = \cos\theta + i\sin\theta$$

As per lecture notes, the State Info Card in the QUI gives the value of the complex amplitude a_ϕ of a given basis state component $|\phi\rangle = |0\rangle, |1\rangle, |00\rangle, |01\rangle, \ldots$ is given by $a_\phi = |a_\phi|e^{i\theta_\phi}$ where $|a_\phi|$ is the magnitude, and θ_ϕ is the phase angle (colour wheel scale). The probability of measuring the state $|\phi\rangle$ is $|a_\phi|^2$.

In the following exercises we will compute some further examples by hand and compare the QUI output.

Another useful QUI feature to note for some of the exercises to follow is that in single qubit mode hovering the mouse over each gate in the programmed circuit will display the Bloch sphere rotation corresponding to that operation on the state at that point.



Exercise 1.5.1 Compute by hand the single gate operations H, X, Y, Z, S, and T on the state $|0\rangle$, and complete the table below. Compare with the QUI in each case.

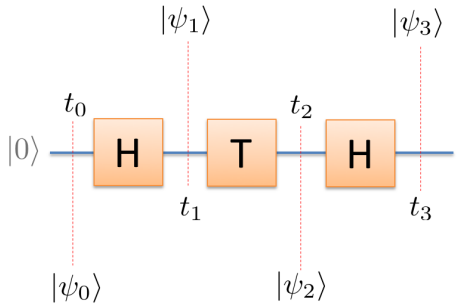
Gate	Operator (matrix rep)	Operation (matrix rep)	Operation (ket rep)	Final state $\|0\rangle$ amplitude	Final state $\|1\rangle$ amplitude	Final state probabilities
H	$\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}$	$\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix}$ $= \frac{1}{\sqrt{2}}\begin{pmatrix}1\\ 1\end{pmatrix}$	$H\|0\rangle$ $= \frac{1}{\sqrt{2}}\|0\rangle + \frac{1}{\sqrt{2}}\|1\rangle$ $= \|a_0\|e^{i\theta_0}\|0\rangle + \|a_1\|e^{i\theta_1}\|1\rangle$	$\|a_0\| = \frac{1}{\sqrt{2}}$ $\theta_0 = 0$	$\|a_1\| = \frac{1}{\sqrt{2}}$ $\theta_1 = 0$	$p_0 = 0.5$ $p_1 = 0.5$
X	$\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix}$ $= \begin{pmatrix}0\\ 1\end{pmatrix}$	$X\|0\rangle = \|1\rangle$	$\|a_0\| = 0$ $\theta_0 = 0$	$\|a_1\| = 1$ $\theta_1 = 0$	$p_0 = 0$ $p_1 = 1$
Y	$\begin{pmatrix}0 & -i\\ i & 0\end{pmatrix}$					
Z	$\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}$					
S	$\begin{pmatrix}1 & 0\\ 0 & i\end{pmatrix}$					
T	$\begin{pmatrix}1 & 0\\ 0 & e^{i\pi/4}\end{pmatrix}$					

Exercise 1.5.2 Compute by hand the single gate operations H, X, Y, Z, S, and T on the state |1⟩, and complete the table below. Compare with the QUI in each case.

Gate	Operator (matrix rep)	Operation (matrix rep)	Operation (ket rep)	Final state $\|0\rangle$ amplitude	Final state $\|1\rangle$ amplitude	Final state probabilities
H	$\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}$	$\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}0\\ 1\end{pmatrix}$ $= \frac{1}{\sqrt{2}}\begin{pmatrix}1\\ -1\end{pmatrix}$	$H\|1\rangle$ $= \frac{1}{\sqrt{2}}\|0\rangle - \frac{1}{\sqrt{2}}\|1\rangle$ $= \|a_0\|e^{i\theta_0}\|0\rangle + \|a_1\|e^{i\theta_1}\|1\rangle$	$\|a_0\| = \frac{1}{\sqrt{2}}$ $\theta_0 = 0$	$\|a_1\| = \frac{1}{\sqrt{2}}$ $\theta_1 = \pi$	$p_0 = \frac{1}{2}$ $p_1 = \frac{1}{2}$
X	$\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix}$ $= \begin{pmatrix}0\\ 1\end{pmatrix}$	$X\|1\rangle = \|0\rangle$	$\|a_0\| = 1$ $\theta_0 = 0$	$\|a_1\| = 0$ $\theta_1 = 0$	$p_0 = 1$ $p_1 = 0$
Y	$\begin{pmatrix}0 & -i\\ i & 0\end{pmatrix}$					
Z	$\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}$					
S	$\begin{pmatrix}1 & 0\\ 0 & i\end{pmatrix}$					
T	$\begin{pmatrix}1 & 0\\ 0 & e^{i\pi/4}\end{pmatrix}$					

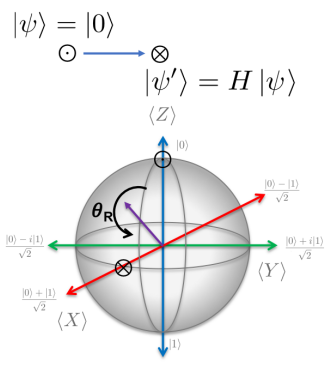
1.6 Sequential gates

Exercise 1.6.1 Program the following sequence of single qubit gates H-T-H (shown right). Compute by hand the states at each time step in the matrix representation, covert to ket representation and fill out the table below. Now compare the amplitudes you obtained with the QUI output at each time step and check they agree.

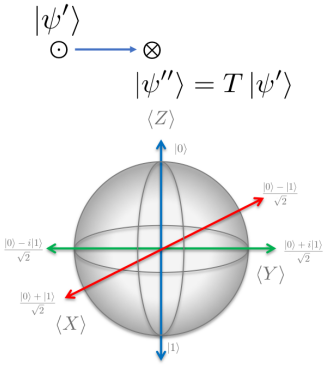


Matrix representation	Ket representation	Amplitudes
$\|\psi_0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$	$\|\psi_0\rangle = \|0\rangle$	$\|a_0\| = 1 \quad \|a_1\| = 0$ $\theta_0 = 0 \quad \theta_1 = 0$
$\|\psi_1\rangle = H\|\psi_0\rangle$		
$\|\psi_2\rangle = T\|\psi_1\rangle$		
$\|\psi_3\rangle = H\|\psi_2\rangle$		

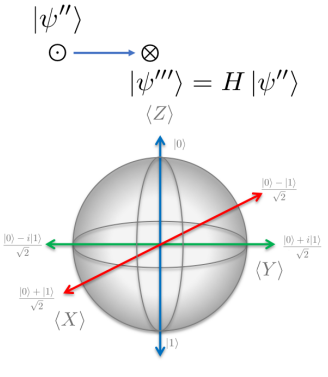
Exercise 1.6.2 Examine the QUI Bloch sphere animations and complete the following Bloch sphere representations of these gates in the sequence H-T-H as per below:



H-gate



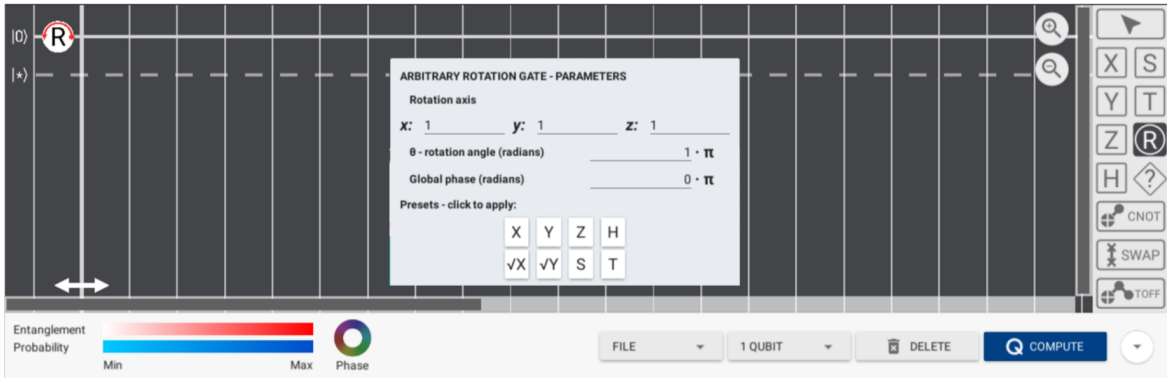
T-gate



H-gate

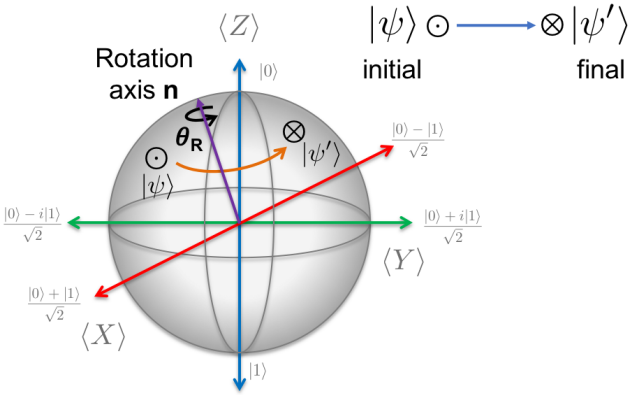
1.7 Arbitrary rotation and global phase

To program the arbitrary rotation gate, R, place it on a single qubit in QUI and right click on the gate. Now select "edit parameters" option and the pop-up window will appear:



In this window, you can select rotation axis (will self-normalise), angle, and global phase.

For reference, the R-gate is defined as (refer to Lecture 2):



$$|\psi'\rangle = R_{\hat{\mathbf{n}}}(\theta_{\mathrm{R}})\,|\psi\rangle$$

$$R_{\hat{\mathbf{n}}}(\theta_{\mathrm{R}}) = \mathbf{1}\cos\frac{\theta_{\mathrm{R}}}{2} - i\,\hat{\mathbf{n}}\cdot\mathbf{X}\sin\frac{\theta_{\mathrm{R}}}{2}$$

$$\mathbf{X} = (X, Y, Z) \qquad \hat{\mathbf{n}} = \frac{\mathbf{n}}{|\mathbf{n}|}$$

Exercise 1.7.1 Here we will note the difference between Bloch rotation angle and amplitude phase angle. In the arbitrary rotation gate menu set the axis to X and $\theta_R = \pi/3$ (use sufficient decimal places). The state created is:

$$|0\rangle \rightarrow \frac{\sqrt{3}}{2}\,|0\rangle + \frac{-i}{2}\,|1\rangle = \qquad |0\rangle + \qquad |1\rangle$$

Write this above (space provided) in terms of the complex amplitudes in polar form, $|a_0|e^{i\theta_0}|0\rangle + |a_1|e^{i\theta_1}|1\rangle$, and verify this by examining the SICs in the QUI output.

Exercise 1.7.2 Using the arbitrary rotation gate, R, program a rotation which is exactly equivalent to the Hadamard gate rotation (i.e. by setting the global phase appropriately). Confirm that it produces the correct final states when acting on both computational states.

Exercise 1.7.3 Program HTH using R-gates with global phase set to zero. Compare the final state with that of **Ex 1.6.1** and verify that the probabilities are unaffected by the choice of global phase. Find and set the global phase for each R-gate to exactly match the final phase of the HTH circuit.

Answer: Global phase (R = H-gate) = _____ ; Global phase (R = T-gate) = _____

Exercise 1.7.4 Add a measurement gate at the end of the HTH sequence. Hit the compute button many times (N) and record the number of 0 and 1 outcomes and fill in the table below. Compare the estimated probabilities with those expected.

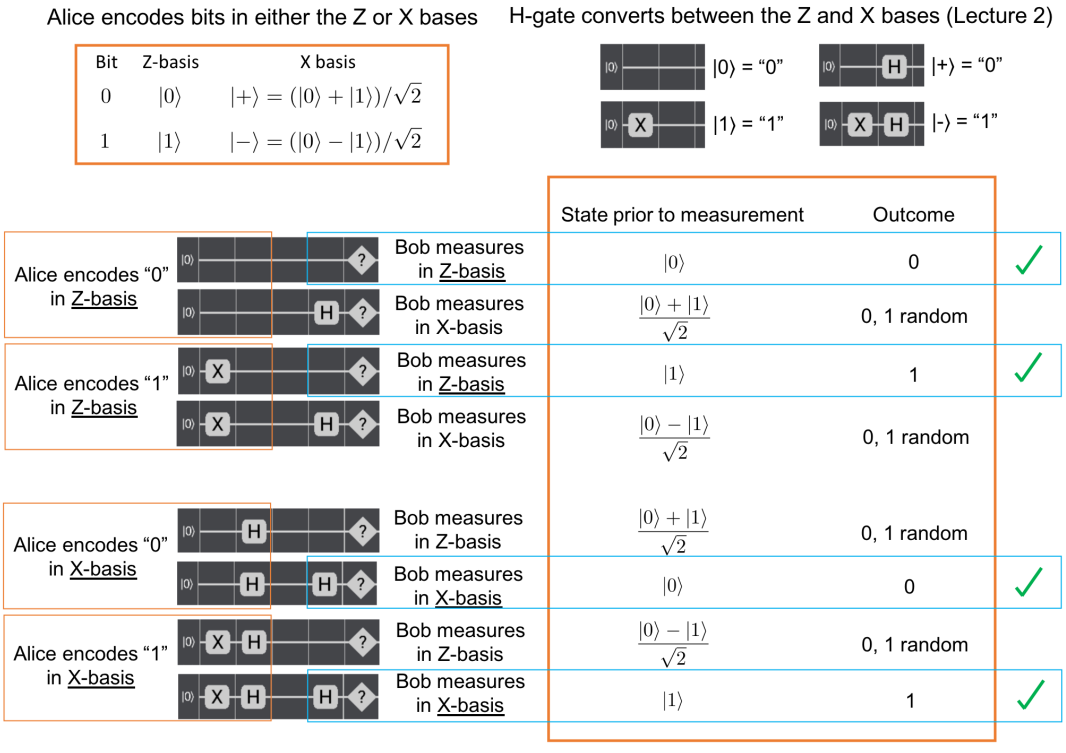
$\|\psi\rangle = HTH\,\|0\rangle$ components	Exact probability	# outcomes, n	Estimated Prob = n/N
$\|0\rangle$	0.854		
$\|1\rangle$	0.146		

1.8 Quantum Communication – the BB84 Protocol in the QUI

An important application of qubits is to enable secure communication between two parties. There are several different quantum communication protocols with varying degrees of complexity. In this section, we will implement a very simple protocol known as "BB84" which is based on single qubits and can be implemented in the QUI. BB84 stands for Bennett and Brassard, the names of the two mathematicians who proposed this protocol in 1984.

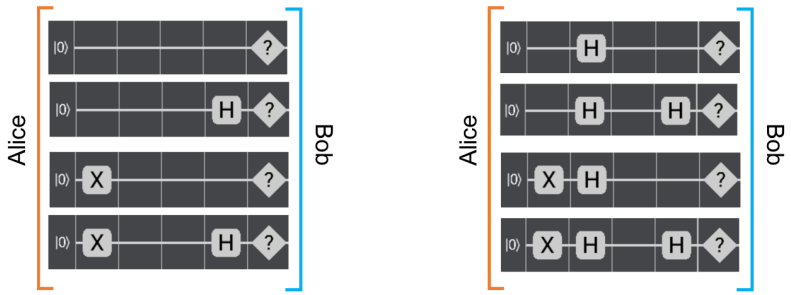
Basic idea: Alice wants to send Bob a secret key (string of bits) over a channel that might be intercepted by an eavesdropper (Eve). Using qubits Alice can encode her bits in randomly selected instances of either the Z basis ($|0\rangle$, $|1\rangle$) or X basis ($|+\rangle$, $|-\rangle$). By the rules of quantum superposition, if Alice prepares a bit in one basis and Bob measures in a different basis he is not guaranteed of measuring the correct bit of information. Only when the bases agree will Bob measure Alice's bit with 100% certainty (see below). The trick is that after measurement Alice and Bob only compare measurement bases, not the actual bits measured. They then use a certain number of correct bits to test whether an eavesdropper tried to intercept and re-send the communication – in this case Alice and Bob will see the percentage of correct bits drop.

Key points of BB84: Bit coding, basis, and measurement



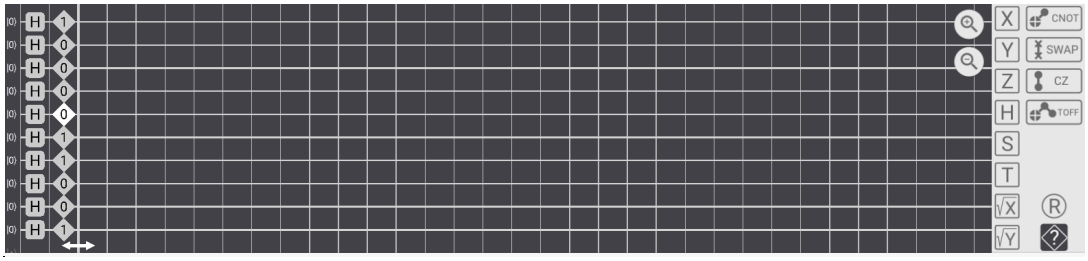
-> Bob obtains the correct bit when he measures in the same basis that Alice uses to encode.

Exercise 1.8.1 For a single qubit, code the BB84 protocol for the no-eavesdropper case in the QUI and repeat for all eight cases of bases choice by Alice and Bob (illustrated below). In single qubit mode, use the time scrubber to examine the qubit state during the protocol to make sure you understand how it works.



Before we start encoding BB84 in the QUI, let's first do a few simple tasks which will allow us to implement the protocol.

Exercise 1.8.2 Generate random binary sequences of (10-bit length) using a Hadamard gate followed by a measurement gate. Recall: a H gate places the state into an equal superposition of |0⟩ and |1⟩, and the measurement will randomly collapse it to either |0⟩ or |1⟩. Construct the following ten qubit circuit in QUI:



Press the COMPUTE button (7 times) to produce the following series of 10-bit random sequences and record as per below:

No-eavesdropper case:

Sequence 1: | | | | | | | | | | | (Alice's bit string)

Sequence 2: | | | | | | | | | | | (Alice's basis choice)

Sequence 3: | | | | | | | | | | | (Bob's basis choice)

Eavesdropper case:

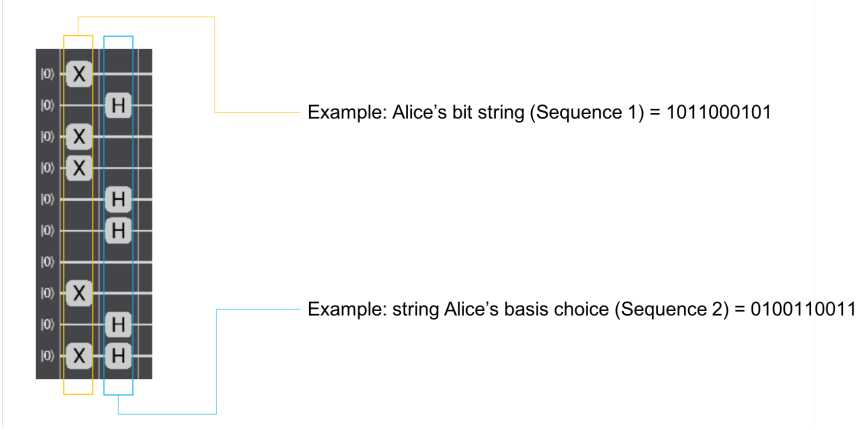
Sequence 4: | | | | | | | | | | | (Alice's bit string)

Sequence 5: | | | | | | | | | | | (Alice's basis choice)

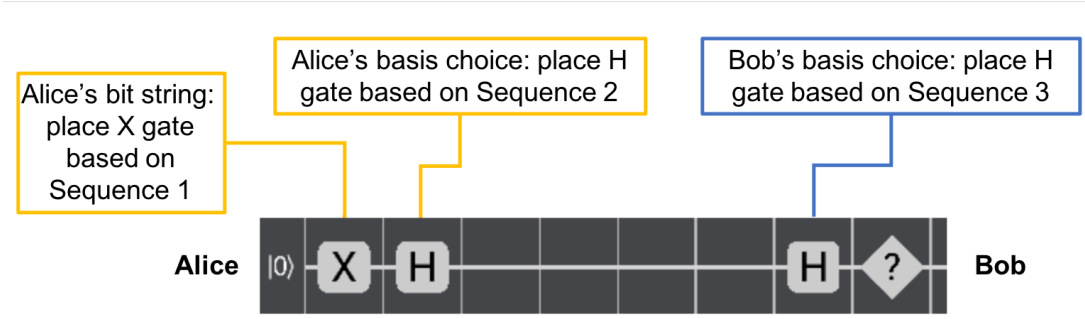
Sequence 6: | | | | | | | | | | | (Bob's basis choice)

Sequence 7: | | | | | | | | | | | (Eve's basis choice)

Exercise 1.8.3 Implement a 10-qubit circuit in QUI to encode the 10-bit binary "Sequence 1" as per the above, corresponding to Alice's bit string. Now use Sequence 2 corresponding to Alice's random choice of basis encoding: "0" for the Z-basis (no H gate), "1" for the X-basis (H-gate), As an example, a circuit is shown below for Sequence 1=1011000101 and Sequence 2=0100110011:



Exercise 1.8.4 Code the BB84 protocol for the no-eavesdropper case for all 10 qubits according to your random sequences 1-3 (as per illustration below).



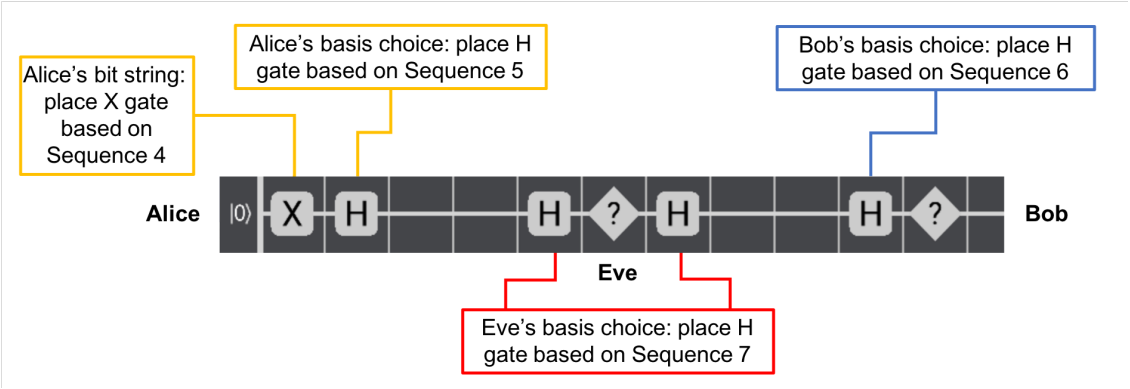
Run the no-eavesdropper-case protocol and fill in the following table:

Table 1.8.1 – No eavesdropper case

Alice sends (0, 1)	Alice basis (X, Z)	Bob basis (X, Z)	Bob receives (0, 1)	Alice-Bob: correct basis? (yes, no)	Correct bit received? (yes, no)	Kept Bits (0, 1, -)

Number of correct bits received = ____ (= ____ %)

Exercise 1.8.5 Now add Eve to your circuit, using the sequences 4-7 as indicated below.



Run the eavesdropper-case protocol and fill in the following table:

Table 1.8.2 – Eavesdropper case

Alice sends (0, 1)	Alice basis (X, Z)	Eve's basis (X, Z)	Eve receives (0, 1)	Bob basis (X, Z)	Bob receives (0, 1)	Alice-Bob: correct basis? (yes, no)	Correct bit received? (yes, no)

Number of correct bits received = ____ (= ____ %)

1.9 Putting it all together

In the presence of Eve the number of correct bits sent is reduced due to her interference, and if there was no other "noise" in the channel that alone would signal an attack. Without Eve, Alice and Bob would agree in basis choice 50% of the time, and hence 50% of the bits sent can be used for the key. With Eve attacking, the percentage of correct bits will be reduced - the presence of Eve can be detected by comparing the percentage of correct bits for a sub-set of the original string (that will never be used for the key).

Exercise 1.9.1 Compare the % correct bits received for both the no-Eve and Eve cases. Each student will only have a small data set, hence we will sum the results and tally (assuming enough people have reached this far!).