

# Quantum Cube Model LaTeX Package (v0.2.0)

Cedric Schacht  
cedric.schacht@dhbw-stuttgart.de

December 19, 2025

## Introduction

The **Quantum Cube Model (QCM)** package provides commands that make it easy to create diagrams representing the quantum cube model for up to 3 qubits. The package simplifies drawing complex quantum state diagrams inspired by Prof. B. Just's framework.

The key difference to version v0.1.0 of this package is, the new version exposes the tikz picture canves to the user. This allows for far more refined diagramms, and flexibility.

## Requirements

To use this package include `\usepackage{quantumcubemodel}` in your documents preamble. This package depends on the following LaTeX packages:

```
\RequirePackage{kvoptions}  
\RequirePackage{braket}  
\RequirePackage{xcolor}  
\RequirePackage{tikz}  
\usetikzlibrary{3d, calc, arrows.meta}
```

## License

MIT License  
Copyright (c) 2025 Cedric Schacht

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## References

Bettina Just. *Quantum Computing Compact: Spooky Action at a Distance and Teleportation Easy to Understand*. Berlin, Heidelberg: Springer, 2022. ISBN: 978-3-662-65007-3 978-3-662-65008-0. DOI: 10.1007/978-3-662-65008-0. URL: <https://link.springer.com/10.1007/978-3-662-65008-0> (visited on 05/26/2025)

## Changelog

- Provide qcmx environment that exposes tikz canvas.
- Give an option to set the main color.
- Provide commands for easier amplitude configuration.
- Provide options to change qubit axis.
- Enable the use of multiple qcm diagrams in one tikz canvas.

## Package options

To change the appearance of your diagrams you can set the main color as follows:

```
\documentclass[preview, border=40px]{standalone}
\usepackage[maincolor=orange]{../quantumcubemodel}

\begin{document}
\begin{qcmx}
  \qcmxO{0.71}
  \qcmxI{0.71}
  \qcmxRenderQ
\end{qcmx}
\end{document}
```



## Provided commands

All commands are prefixed with **qcmx**.

There are other commands that start with **QCMXINTERNAL** these are for internal purposes only, the use of those is not recommended.

```
\begin{qcmx}
  % Everything happens in this environment
\end{qcmx}
```

In this environment you can call one of the following commands to render a qcm diagram for one, two or three qubit diagrams respectively:

- `\qcmxRenderQ`
- `\qcmxRenderQQ`
- `\qcmxRenderQQQ`

Initially all amplitudes are set to 0, you can change them with the commands of the following pattern:

`\qcmxO{}` or `\qcmxI{}` for one qubit amplitude at  $|0\rangle$  (O) or  $|1\rangle$  (I).

`\qcmxOO{}`, `\qcmxOI{}`, `\qcmxIO{}` and `\qcmxII{}` for systems with two qubits.

`\qcmxOOO{}` up to `\qcmxIII{}` for systems with three qubits..

You have to set the amplitudes before calling the corresponding render command.

Each of these commands gives you an optional parameter to set a phase (in degrees from 0 to 360) for the coefficient, for the state  $-|+\rangle$  this results in:

```
\begin{qcmx}
  \qcmxO[180]{0.71}
  \qcmxI[180]{0.71}
  \qcmxRenderQ
\end{qcmx}
```

If you want to include more than one qcm diagram in one figure, you can define `\def\qcmxOffsetX{5}` to render the next one 5 units to the left. There are offsets for X, Y and Z where the direction follows the tikz orientations. To reset the amplitudes and phases back to 0, you can use the corresponding reset command:

- `\qcmxClearQ`
- `\qcmxClearQQ`
- `\qcmxClearQQQ`

Use `\def\qcmxOrientationQ{y}` to set the orientation of a one qubit system to the y-axis. This needs to be defined before using the render command. The commands for multi qubit systems follow the scheme of Q, QQ and QQQ like the render commands. Possible values for orientations are:

- in Systems with one qubit: x, y, z (default: x)
- in Systems with two qubits: xy, xz, yz (default: xy)
- in Systems with three qubits: xyz (default: xyz)

For the transformations there are specialized render commands for the Pauli-Gates X, Y, Z as well as the Hadamard transformation. In multi qubit systems additionally there are renderers for CNot and CCNot (Toffoli). These follow a scheme like this: `\qcmxRender[OperationName][QubitCount]{qubits}` where `[OperationName]` can be i.e. Hadamard, the `[QubitCount]` can be QQ.

To render the hadamard transformation on the x qubit in a system with two qubits this results in:

```
\qcmxRenderHadamardQQ{x}
```

Keep in mind that the `\qcmxOrientationQQ` setting will be used to determine that the other qubit is y.

```
\def\qcmxOrientationQQ{xz}
\qcmxRenderHadamardQQ{x}
```

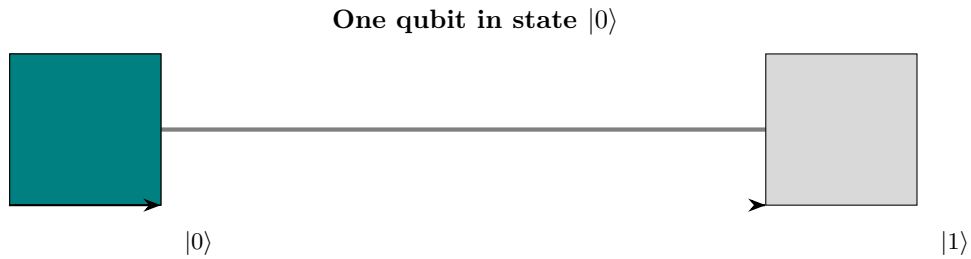
Results in a different diagram.

The same is true for measurements, the only exceptions are CNot and CCNot, where there is no single qubit (or two qubit for CCNot) version.

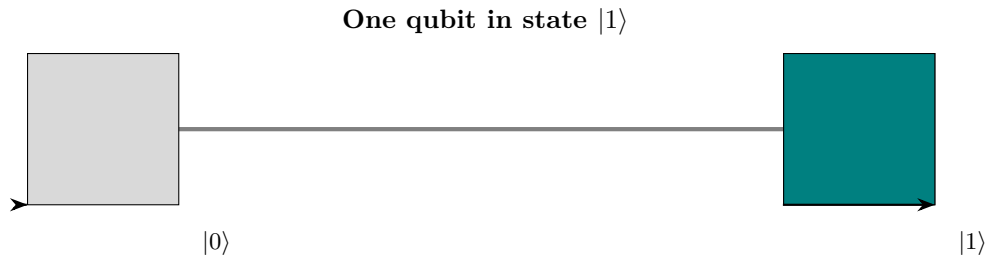
- `\qcmxRenderHadamardQ` one of [x,y,z]
- `\qcmxRenderHadamardQQ` one of [x,y,z]
- `\qcmxRenderHadamardQQQ` one of [x,y,z]
- `\qcmxRenderPauliXQ` one of [x,y,z]
- `\qcmxRenderPauliXQQ` one of [x,y,z]
- `\qcmxRenderPauliXQQQ` one of [x,y,z]
- `\qcmxRenderPauliYQ` one of [x,y,z]
- `\qcmxRenderPauliYQQ` one of [x,y,z]
- `\qcmxRenderPauliYQQQ` one of [x,y,z]
- `\qcmxRenderPauliZQ` one of [x,y,z]
- `\qcmxRenderPauliZQQ` one of [x,y,z]
- `\qcmxRenderPauliZQQQ` one of [x,y,z]
- `\qcmxRenderCNotQQ` one of [xy,yx,xz,zx,yz,zy]
- `\qcmxRenderCNotQQQ` one of [xy,yx,xz,zx,yz,zy]
- `\qcmxRenderCCNotQQQ` one of [xyz,yxz,xzy,zxy,yzx,zyx]
- `\qcmxRenderMeasureQ` one of [x,y,z]
- `\qcmxRenderMeasureQQ` one of [x,y,z,xy,xz,yz]
- `\qcmxRenderMeasureQQQ` one of [x,y,z,xy,xz,yz,xyz]

## One qubit states

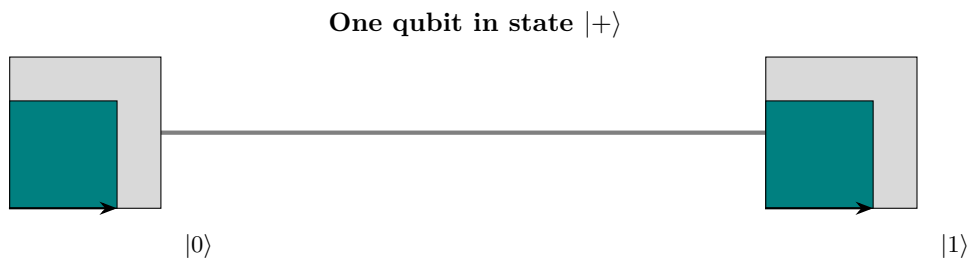
```
\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\textbf{One qubit in state  $|\ket{0}\rangle$ }
\begin{qcmx}
\qcmxO{1}
\qcmxRenderQ
\end{qcmx}
\end{document}
```



```
\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\textbf{One qubit in state  $|\ket{1}\rangle$ }
\begin{qcmx}
\qcmxI{1}
\qcmxRenderQ
\end{qcmx}
\end{document}
```



```
\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\textbf{One qubit in state  $|\ket{+}\rangle$ }
\begin{qcmx}
\qcmxO{0.71}
\qcmxI{0.71}
\qcmxRenderQ
\end{qcmx}
\end{document}
```



## Amplitudes & Phases

```
\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\textbf{One qubit in a state with arbitrary amplitudes}
\begin{qcmx}
  \qcmxO{0.85}
  \qcmxI{0.53}
  \qcmxRenderQ
\end{qcmx}
\end{document}
```

**One qubit in a state with arbitrary amplitudes**



```
\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\textbf{One qubit in a state with complex phases}
\begin{qcmx}
  \qcmxO[30]{0.85}
  \qcmxI[145]{0.53}
  \qcmxRenderQ
\end{qcmx}
\end{document}
```

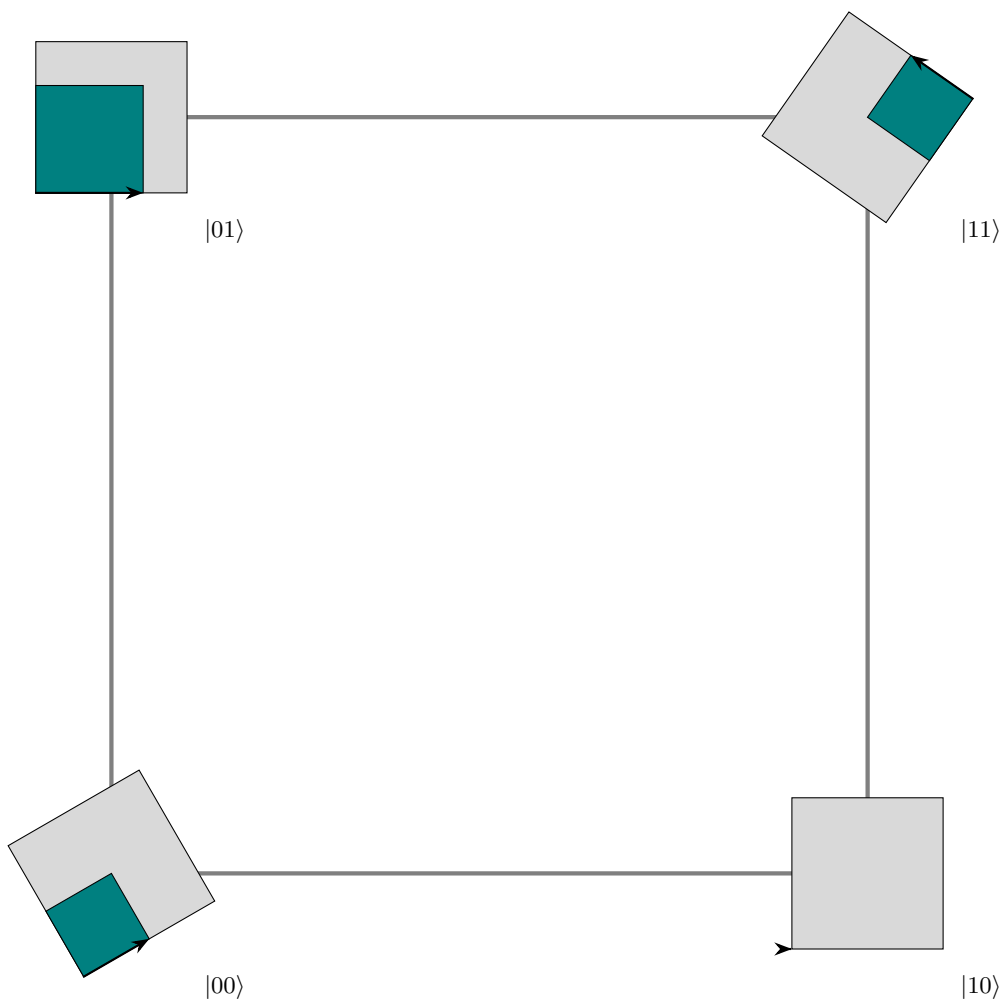
**One qubit in a state with complex phases**



## Two qubit state

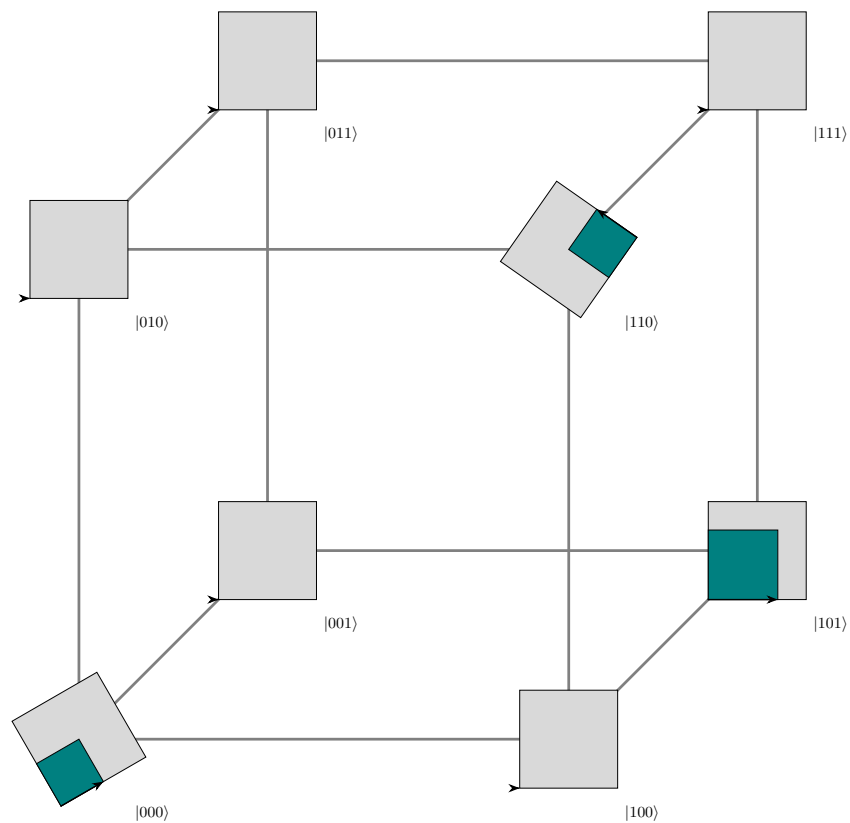
```
\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\textbf{Two qubits in a state with arbitrary amplitudes and phases}
\begin{qcmx}
\qcmxOO[30]{0.5}
\qcmxOI{0.71}
% \qcmxIO{0} % can be left out, since this is the default value
\qcmxII[145]{0.5}
\qcmxRenderQQ
\end{qcmx}
\end{document}
```

Two qubits in a state with arbitrary amplitudes and phases



## Three qubit state

```
\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\begin{qcmx}
  \qcmxOOO[30]{0.5}
  % \qcmxOOI{0} % can be left out, since this is the default value
  % \qcmxOIO{0}
  % \qcmxOII{0}
  % \qcmxIOO{0}
  \qcmxIOI{0.71}
  \qcmxIIO[145]{0.5}
  % \qcmxIII{0} % can be left out, since this is the default value
  \qcmxRenderQQQ
\end{qcmx}
\end{document}
```





## Orientation and Positioning of qubits

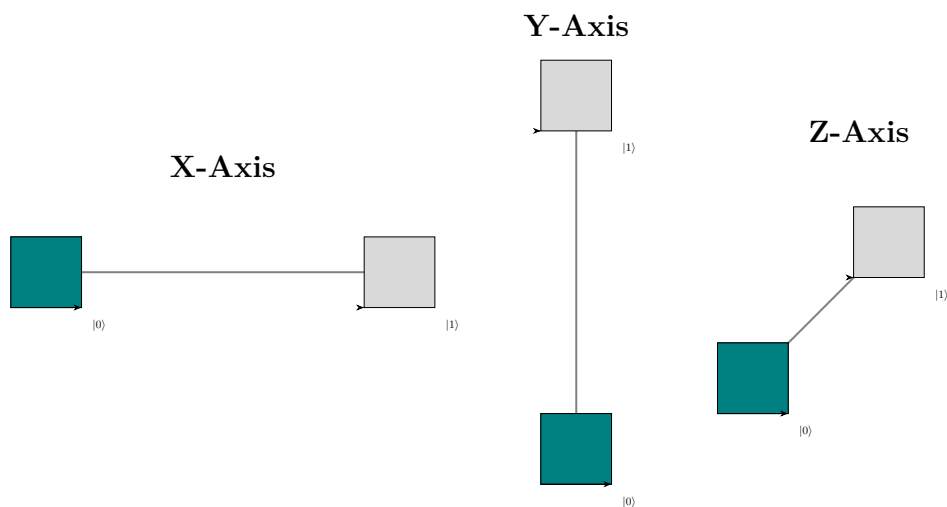
```
\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\begin{qcmx}
  \node at (5, 3, 0) {\Huge \textbf{X-Axis}};
  \def\qcmxOrientationQ{x} % this is the default
  \qcmxO{1}
  \qcmxRenderQ

  \node at (15, 7, 0) {\Huge \textbf{Y-Axis}};
  \def\qcmxOffsetX{15} % <— this moves the Y-Axis qubit to the right
  \def\qcmxOffsetY{-5} % and a bit down, relative to the origin (0, 0, 0)

  \def\qcmxOrientationQ{y} % <— this sets the orientation to the y axis
  \qcmxRenderQ

  \node at (23, 4, 0) {\Huge \textbf{Z-Axis}};
  \def\qcmxOffsetX{20}
  \def\qcmxOffsetY{-3}

  \def\qcmxOrientationQ{z}
  \qcmxRenderQ
\end{qcmx}
\end{document}
```



The same works for:

```
\def\qcmxOrientationQQ{xy}
```

Possible values are xy, xz, yz to set the plane accordingly.

And for future use: `\def\qcmxOrientationQQQ{xyz}` is reserved, but there are no other valid values except xyz for now.

As you can see, the qcmx environment exposes a 3d Tikz canvas, there for you are free to use commands like `\node at...` to add descriptive markings.

## Two qubit state from two one qubit state

```

\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\begin{qcmx}
  \qcmxO{0.71}
  \qcmxI{0.71}
  \qcmxRenderQ
  \qcmxClearQ

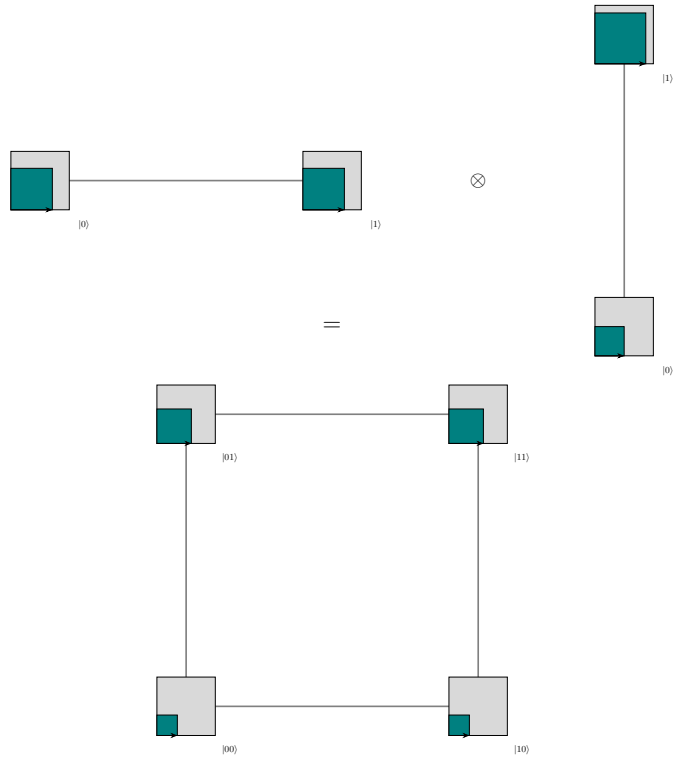
  \node at (15,0,0) {\huge$\otimes$};

  \qcmxO{0.5}
  \qcmxI{0.87}
  \def\qcmxOffsetX{20}
  \def\qcmxOffsetY{-5}
  \def\qcmxOrientationQ{y}
  \qcmxRenderQ
  \qcmxClearQ

  \node at (10,-5,0) {\Huge$=$};

  \qcmxOO{0.35}
  \qcmxOI{0.59}
  \qcmxIO{0.35}
  \qcmxII{0.59}
  \def\qcmxOffsetX{5}
  \def\qcmxOffsetY{-18}
  \qcmxRenderQQ
\end{qcmx}
\end{document}

```



## Transformation diagrams

```

\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\begin{qcmx}
  \node at (10, 0, 0) {\Huge \textbf{Hadamard transformation on one qubit}};
  \def\qcmxOffsetX{0}
  \def\qcmxOffsetY{-3}
  \qcmxO{1}
  \qcmxRenderQ
  \qcmxClearQ

  \def\qcmxOffsetX{15}
  \qcmxRenderHadamardQ{x}

  \qcmxO{0.71}
  \qcmxI{0.71}
  \def\qcmxOffsetX{25}
  \qcmxRenderQ
  \qcmxClearQ

  \node at (10,-10, 0) {\Huge \textbf{Pauli-X transformation on one qubit}};
  \def\qcmxOffsetX{0}
  \def\qcmxOffsetY{-13}
  \qcmxO{1}
  \qcmxRenderQ
  \qcmxClearQ

  \def\qcmxOffsetX{15}
  \qcmxRenderPauliXQ{x}

  \qcmxI{1}
  \def\qcmxOffsetX{25}
  \qcmxRenderQ
  \qcmxClearQ

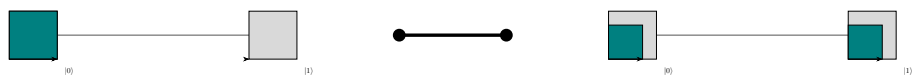
  \node at (10,-20, 0) {\Huge \textbf{Pauli-Y and Z transformation on one qubit}};
  \def\qcmxOffsetX{0}
  \def\qcmxOffsetY{-24}
  \qcmxO{0.71}
  \qcmxI{0.71}
  \qcmxRenderQ
  \qcmxClearQ

  \def\qcmxOffsetX{15}
  \def\qcmxOffsetY{-23}
  \qcmxRenderPauliYQ{x}
  \def\qcmxOffsetY{-25}
  \qcmxRenderPauliZQ{x}

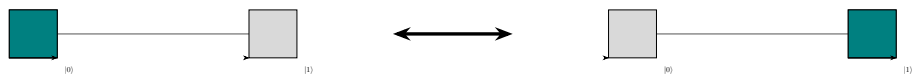
  \def\qcmxOffsetY{-24}
  \qcmxO[90]{0.71}
  \qcmxI[90]{0.71}
  \def\qcmxOffsetX{25}
  \qcmxRenderQ
\end{qcmx}
\end{document}

```

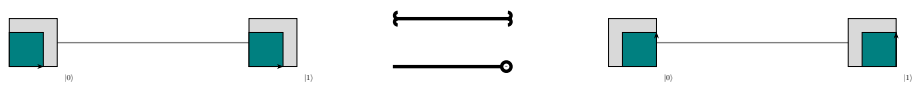
### Hadamard transformation on one qubit



### Pauli-X transformation on one qubit



### Pauli-Y and Z transformation on one qubit



```

\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\begin{qcmx}
  \node at (18, 13, 0) {\Huge \textbf{Hadamard on first of two qubits in xy plane}};
  \qcmxOO{1}
  \qcmxRenderQQ
  \qcmxClearQQ

  \def\qcmxOffsetX{15}
  \def\qcmxOffsetY{3}
  \qcmxRenderHadamardQQ{x}

  \def\qcmxOffsetX{25}
  \def\qcmxOffsetY{0}
  \qcmxOO{0.71}
  \qcmxIO{0.71}
  \qcmxRenderQQ
  \qcmxClearQQ

  \node at (18, -7, 0) {\Huge \textbf{Hadamard on second of two qubits in xy plane}};
  \def\qcmxOffsetX{0}
  \def\qcmxOffsetY{-20}
  \qcmxOO{1}
  \qcmxRenderQQ
  \qcmxClearQQ

  \def\qcmxOffsetX{15}
  \def\qcmxOffsetY{-17}
  \qcmxRenderHadamardQQ{y}

  \def\qcmxOffsetX{25}
  \def\qcmxOffsetY{-20}
  \qcmxOO{0.71}
  \qcmxOI{0.71}
  \qcmxRenderQQ
  \qcmxClearQQ

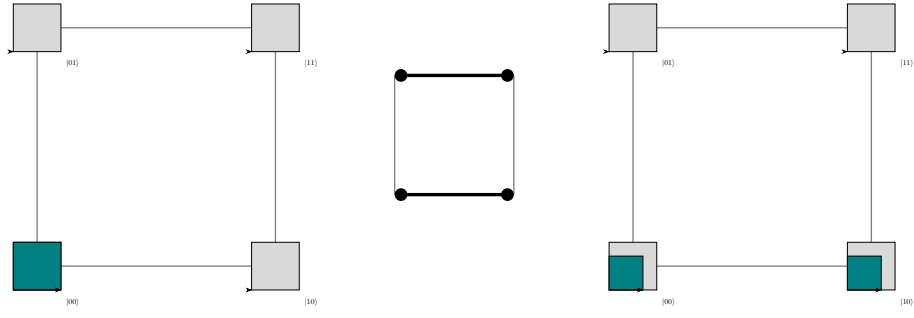
  \node at (18, -27, 0) {\Huge \textbf{Hadamard on first of two qubits in yz plane}};
  \def\qcmxOrientationQQ{yz}
  \def\qcmxOffsetX{5}
  \def\qcmxOffsetY{-45}
  \qcmxOO{1}
  \qcmxRenderQQ
  \qcmxClearQQ

  \def\qcmxOffsetX{17}
  \def\qcmxOffsetY{-40}
  \qcmxRenderHadamardQQ{y}

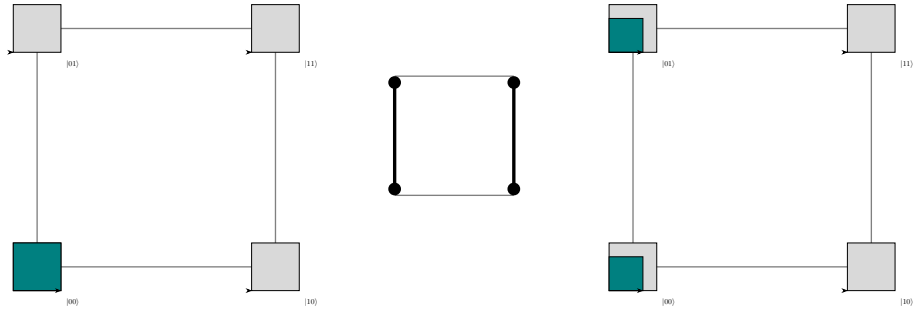
  \def\qcmxOffsetX{27}
  \def\qcmxOffsetY{-45}
  \qcmxOO{0.71}
  \qcmxIO{0.71}
  \qcmxRenderQQ
\end{qcmx}
\end{document}

```

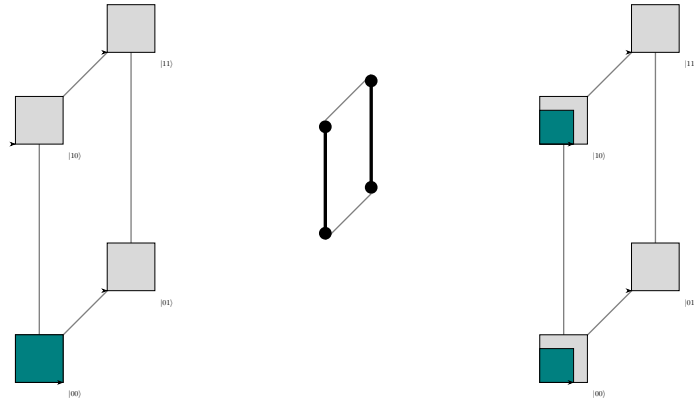
Hadamard on first of two qubits in xy plane



Hadamard on second of two qubits in xy plane



Hadamard on first of two qubits in yz plane



As you can see the orientation of the Hadamard Transform is set by two parts, first the value of `\qcmxOrientationQQ` is used to determine the plane, then the argument of `\qcmxRenderHadamardQQ` is used to select the qubit that is transformed.

The same is true for the pauli gates (x, y, z) as well as the three dimensional cube, even though `\qcmxOrientationQQQ` can only be xyz for now.

## Multi qubit transformations

```

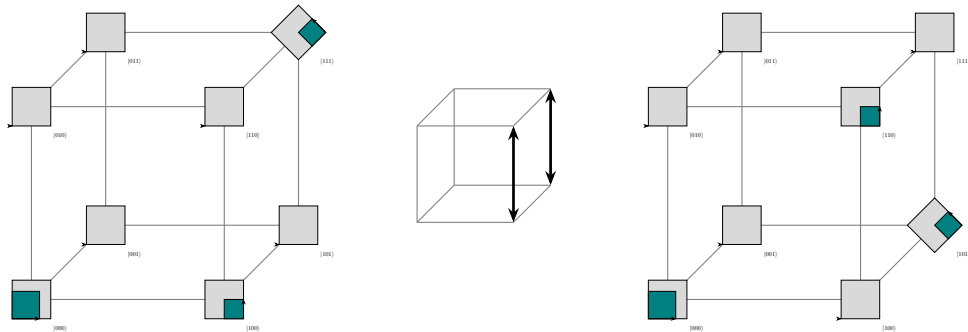
\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\begin{qcmx}
  \node at (24, 18, 0) {\Huge \textbf{
    CNot on x (control), y (target) in a system of three qubits
  }}};
  \qcmxOOO{0.71}
  \qcmxIOO[90]{0.5}
  \qcmxIII[135]{0.5}
  \qcmxRenderQQQ
  \qcmxClearQQQ

  \def\qcmxOffsetX{20}
  \def\qcmxOffsetY{4}
  \qcmxRenderCNotQQQ{xy} % x is the controll, y is the target

  \def\qcmxOffsetX{33}
  \def\qcmxOffsetY{0}
  \qcmxOOO{0.71}
  \qcmxIIIO[90]{0.5}
  \qcmxIOI[135]{0.5}
  \qcmxRenderQQQ
\end{qcmx}
\end{document}

```

CNot on x (control), y (target) in a system of three qubits



Additionally there is the CCNot (Toffoli) gate, that can be rendered with the `\qcmxRenderCCNotQQQ`. Obviously CNot and CCNot are only available in systems with at least two (or three) qubits respectively.

# Measurement of Qubits

```

\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\begin{qcmx}
  \qcmxO{0.71}
  \qcmxI{0.71}
  \qcmxRenderQ
  \qcmxClearQ

  \def\qcmxOffsetX{15}
  \qcmxRenderMeasureQ{x}

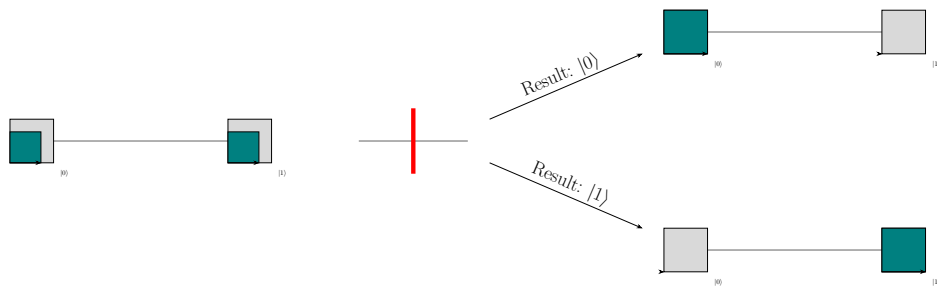
  \draw[black, ->, thick, >=Stealth] (21,1,0) — (28,4,0)
    node[midway, above, sloped] {\Huge Result:  $\ket{0}$ };

  \draw[black, ->, thick, >=Stealth] (21,-1,0) — (28,-4,0)
    node[midway, above, sloped] {\Huge Result:  $\ket{1}$ };

  \def\qcmxOffsetX{30}
  \def\qcmxOffsetY{5}
  \qcmxO{1}
  \qcmxRenderQ
  \qcmxClearQ

  \def\qcmxOffsetX{30}
  \def\qcmxOffsetY{-5}
  \qcmxI{1}
  \qcmxRenderQ
  \qcmxClearQ
\end{qcmx}
\end{document}

```





```

\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\begin{qcmx}
  \qcmxOO{0.5}
  \qcmxOI{0.71}
  \qcmxIO{0.5}
  \qcmxRenderQQ
  \qcmxClearQQ

  \def\qcmxOffsetX{15}
  \def\qcmxOffsetY{3}
  \qcmxRenderMeasureQQ{x}

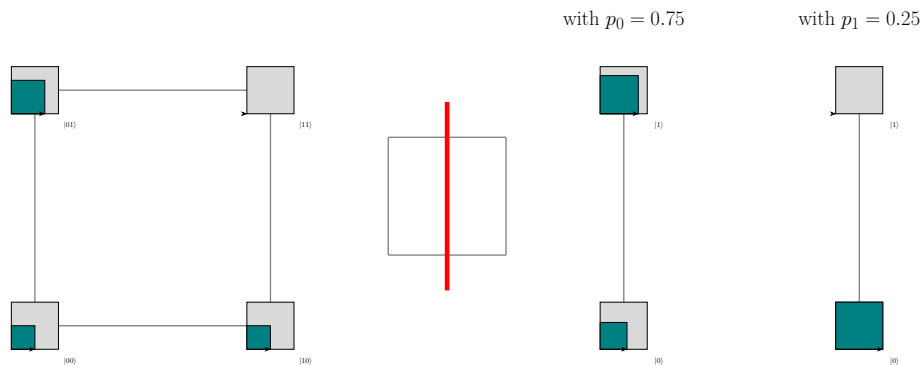
  \def\qcmxOffsetX{25}
  \def\qcmxOffsetY{0}
  \def\qcmxOrientationQ{y}

  \node at (25, 13, 0) {\Huge with  $p_0 = 0.75$ };
  \qcmxO{0.57}
  \qcmxI{0.81}
  \qcmxRenderQ
  \qcmxClearQ

  \node at (35, 13, 0) {\Huge with  $p_1 = 0.25$ };
  \def\qcmxOffsetX{35}
  \qcmxO{1}
  \qcmxRenderQ

\end{qcmx}
\end{document}

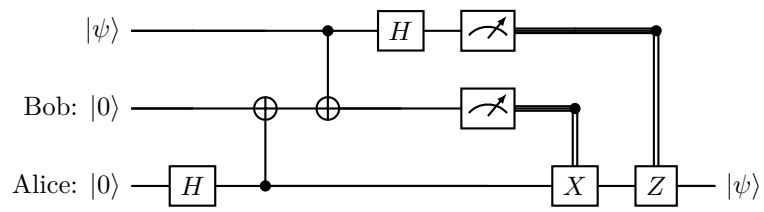
```



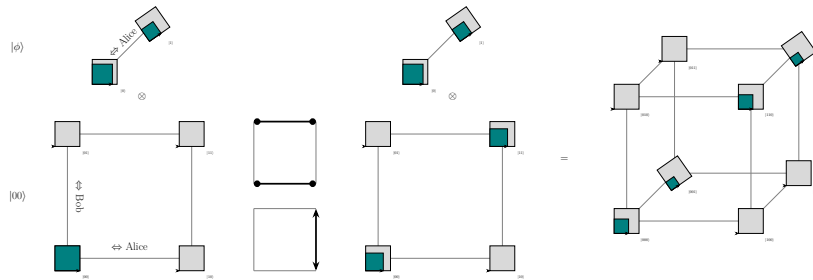
For systems with only partial measurements the `\qcmxOrientationQQ` and `\qcmxOrientationQQQ` values are used to determin the unmeasured qubits.

Additionally it is possible to measure more than one qubit at the same time. For `\qcmxRenderMeasureQQ` it is possible to choose from  $x$ ,  $y$ ,  $z$ ,  $xy$ ,  $xz$ ,  $yz$ . While for `\qcmxRenderMeasureQQQ`  $xyz$  is also available.

# Example: Teleportation Algorithm

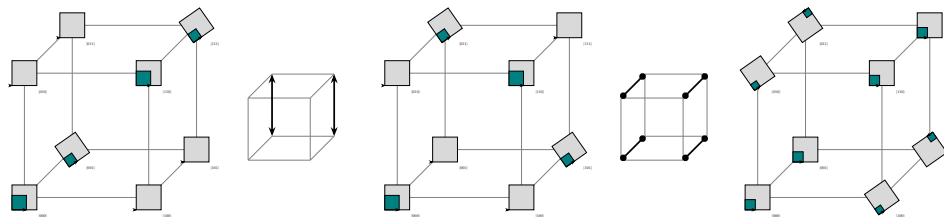


State Preparation

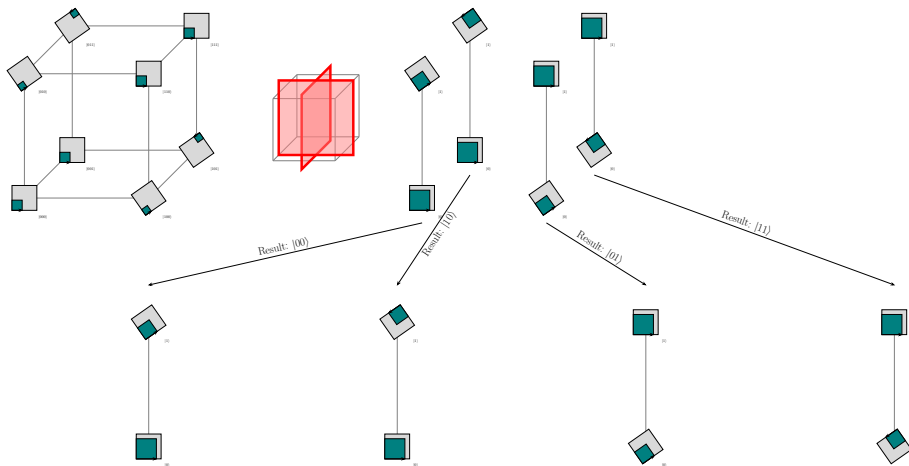


Entangling  $|\phi\rangle$  with EPR

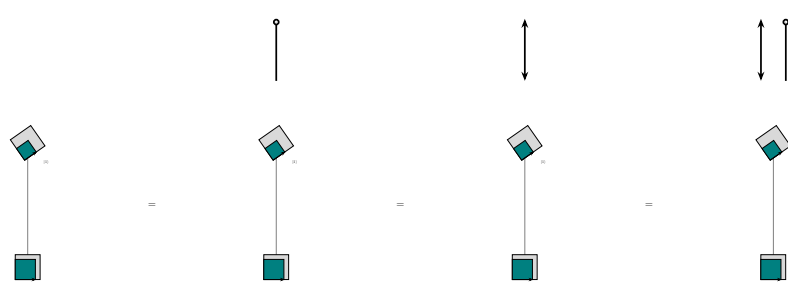
Hadamard transform  $z$



Measuring x and z



Fixing Error



```

\documentclass[preview, border=40px]{standalone}
\usepackage{../quantumcubemodel}
\begin{document}
\begin{qcmx}
  \node at (25,8,0) {\Huge \textbf{State Preparation}};

  \node at (-4,2,0) {\Huge$\ket{\phi}$};
  \node at (-4,-10,0) {\Huge$\ket{00}$};

  \node[rotate=45] at (5,3,1.5) {\Huge $\Leftrightarrow$ Alice};

  \def\qcmxOrientationQ{z}
  \def\qcmxOffsetX{3}
  \def\qcmxOffsetY{0}
  \qcmxO{0.81}
  \qcmxI[35]{0.57}
  \qcmxRenderQ
  \qcmxClearQ
  \node at (6,-2,0) {\Huge$\otimes$};

  \node at (5,-14,0) {\Huge $\Leftrightarrow$ Alice};
  \node[rotate=-90] at (1,-10,0) {\Huge $\Leftrightarrow$ Bob};
  \def\qcmxOffsetX{0}
  \def\qcmxOffsetY{-15}
  \qcmxOO{1}
  \qcmxRenderQQ
  \qcmxClearQQ

  \def\qcmxOffsetX{15}
  \def\qcmxOffsetY{-9}
  \qcmxRenderHadamardQQ{x}
  \def\qcmxOffsetY{-16}
  \qcmxRenderCNotQQ{xy}

  \def\qcmxOffsetX{28}
  \def\qcmxOffsetY{0}
  \qcmxO{0.81}
  \qcmxI[35]{0.57}
  \qcmxRenderQ
  \qcmxClearQ
  \node at (31,-2,0) {\Huge$\otimes$};

  \def\qcmxOffsetX{25}
  \def\qcmxOffsetY{-15}
  \qcmxOO{0.71}
  \qcmxII{0.71}
  \qcmxRenderQQ
  \qcmxClearQQ

  \node at (40,-7,0) {\Huge$==$};
  \def\qcmxOffsetX{45}
  \def\qcmxOffsetY{-12}
  \qcmxOOO{0.58}
  \qcmxOOI[35]{0.4}
  \qcmxIIIO{0.58}
  \qcmxIII[35]{0.4}
  \qcmxRenderQQQ
  \qcmxClearQQQ

```

```

\def\qcmxOffsetX{0}
\def\qcmxOffsetY{-41}
\qcmxOOO{0.58}
\qcmxOOI[35]{0.4}
\qcmxIIO{0.58}
\qcmxIII[35]{0.4}
\qcmxRenderQQQ
\qcmxClearQQQ

\node at (22,-23,0) {\Huge \textbf{Entangling $\ket{\phi}$ with EPR}};
\def\qcmxOffsetX{18}
\def\qcmxOffsetY{-38}
\qcmxRenderCNotQQQ{zy}

\def\qcmxOffsetX{30}
\def\qcmxOffsetY{-41}
\qcmxOOO{0.58}
\qcmxOII[35]{0.4}
\qcmxIIO{0.58}
\qcmxIOI[35]{0.4}
\qcmxRenderQQQ
\qcmxClearQQQ

\node at (51,-23,0) {\Huge \textbf{Hadamard transform $z$}};
\def\qcmxOffsetX{48}
\def\qcmxOffsetY{-38}
\qcmxRenderHadamardQQQ{z}

\def\qcmxOffsetX{59}
\def\qcmxOffsetY{-41}
\qcmxOOO{0.41}
\qcmxOOI{0.41}
\qcmxOIO[35]{0.28}
\qcmxOII[180+35]{0.28}
\qcmxIOO[35]{0.28}
\qcmxIOI[180+35]{0.28}
\qcmxIIO{0.41}
\qcmxIII{0.41}
\qcmxRenderQQQ
\qcmxClearQQQ

\node at (22,-50,0) {\Huge \textbf{Measureing x and z}};
\def\qcmxOffsetX{0}
\def\qcmxOffsetY{-68}
\qcmxOOO{0.41}
\qcmxOOI{0.41}
\qcmxOIO[35]{0.28}
\qcmxOII[180+35]{0.28}
\qcmxIOO[35]{0.28}
\qcmxIOI[180+35]{0.28}
\qcmxIIO{0.41}
\qcmxIII{0.41}
\qcmxRenderQQQ
\qcmxClearQQQ

```

```

\def\qcmxOffsetX{20}
\def\qcmxOffsetY{-65}
\qcmxRenderMeasureQQQ{xz}

\def\qcmxOffsetX{32}
\def\qcmxOffsetY{-68}
\def\qcmxOffsetZ{0}
\def\qcmxOrientationQ{y}
\qcmxO{0.81}
\qcmxI[35]{0.57}
\qcmxRenderQ
\qcmxClearQ

\def\qcmxOffsetX{42}
\def\qcmxOffsetY{-68}
\def\qcmxOffsetZ{0}
\def\qcmxOrientationQ{y}

\qcmxI{0.81}
\qcmxO[35]{0.57}
\qcmxRenderQ
\qcmxClearQ

\def\qcmxOffsetX{32}
\def\qcmxOffsetY{-68}
\def\qcmxOffsetZ{-10}
\def\qcmxOrientationQ{y}

\qcmxO{0.81}
\qcmxI[180+35]{0.57}
\qcmxRenderQ
\qcmxClearQ

\def\qcmxOffsetX{42}
\def\qcmxOffsetY{-68}
\def\qcmxOffsetZ{-10}
\def\qcmxOrientationQ{y}

\qcmxI{0.81}
\qcmxO[180+35]{0.57}
\qcmxRenderQ
\qcmxClearQ

\draw[black, ->, thick, >=Stealth] (32,-70,0) — (10,-75,0)
node[midway, above, sloped] {\Huge Result:  $\ket{00}$ };

\draw[black, ->, thick, >=Stealth] (32,-70,-10) — (30,-75,0)
node[midway, below, sloped] {\Huge Result:  $\ket{10}$ };

\draw[black, ->, thick, >=Stealth] (42,-70,0) — (50,-75,0)
node[midway, above, sloped] {\Huge Result:  $\ket{01}$ };

\draw[black, ->, thick, >=Stealth] (42,-70,-10) — (70,-75,0)
node[midway, above, sloped] {\Huge Result:  $\ket{11}$ };

\node[rotate=90] at (0,-95,0) {\Huge \textbf{Fixing Error}};

```

```

\def\qcmxOffsetX{10}
\def\qcmxOffsetY{-88}
\def\qcmxOffsetZ{0}
\def\qcmxOrientationQ{y}
\qcmxO{0.81}
\qcmxI[35]{0.57}
\qcmxRenderQ
\qcmxClearQ

```

```

\def\qcmxOffsetX{30}
\def\qcmxOffsetY{-88}
\def\qcmxOffsetZ{0}
\def\qcmxOrientationQ{y}
\qcmxO{0.81}
\qcmxI[180+35]{0.57}
\qcmxRenderQ
\qcmxClearQ

```

```

\def\qcmxOffsetX{30}
\def\qcmxOffsetY{-98}
\qcmxRenderPauliZQ{y}

```

```

\def\qcmxOffsetX{50}
\def\qcmxOffsetY{-88}
\def\qcmxOffsetZ{0}
\def\qcmxOrientationQ{y}
\qcmxI{0.81}
\qcmxO[35]{0.57}
\qcmxRenderQ
\qcmxClearQ

```

```

\def\qcmxOffsetX{50}
\def\qcmxOffsetY{-98}
\qcmxRenderPauliXQ{y}

```

```

\def\qcmxOffsetX{70}
\def\qcmxOffsetY{-88}
\def\qcmxOffsetZ{0}
\def\qcmxOrientationQ{y}
\qcmxI{0.81}
\qcmxO[180+35]{0.57}
\qcmxRenderQ
\qcmxClearQ

```

```

\def\qcmxOffsetX{69}
\def\qcmxOffsetY{-98}
\qcmxRenderPauliXQ{y}
\def\qcmxOffsetX{71}
\def\qcmxOffsetY{-98}
\qcmxRenderPauliZQ{y}

```

```

\def\qcmxOffsetX{10}
\def\qcmxOffsetY{-113}
\def\qcmxOffsetZ{0}
\def\qcmxOrientationQ{y}
\qcmxO{0.81}
\qcmxI[35]{0.57}

```

```

\qcmxRenderQ
\qcmxClearQ

\node at (20,-108,0) {\Huge$=$};

\def\qcmxOffsetX{30}
\def\qcmxOffsetY{-113}
\def\qcmxOffsetZ{0}
\def\qcmxOrientationQ{y}
\qcmxO{0.81}
\qcmxI[35]{0.57}
\qcmxRenderQ
\qcmxClearQ

\node at (40,-108,0) {\Huge$=$};

\def\qcmxOffsetX{50}
\def\qcmxOffsetY{-113}
\def\qcmxOffsetZ{0}
\def\qcmxOrientationQ{y}
\qcmxO{0.81}
\qcmxI[35]{0.57}
\qcmxRenderQ
\qcmxClearQ

\node at (60,-108,0) {\Huge$=$};

\def\qcmxOffsetX{70}
\def\qcmxOffsetY{-113}
\def\qcmxOffsetZ{0}
\def\qcmxOrientationQ{y}
\qcmxO{0.81}
\qcmxI[35]{0.57}
\qcmxRenderQ
\qcmxClearQ
\end{qcmx}
\end{document}

```