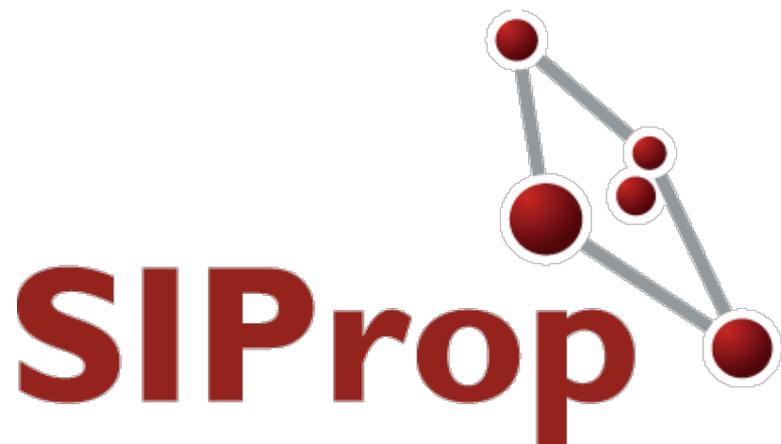


マイクロ波SDRを用いたGNURadioの 超伝導量子ファームウェア拡張について

JI1SZP

今村謙之



● 個人情報

● 今村謙之

● いまむらのりつな

● JI1SZP

● 平28.4.22開局

● 群馬県館林市

● 衛星通信シャック

● 無線機

● IC-9100 + UX-9100

● LimeSDR

● アンテナ

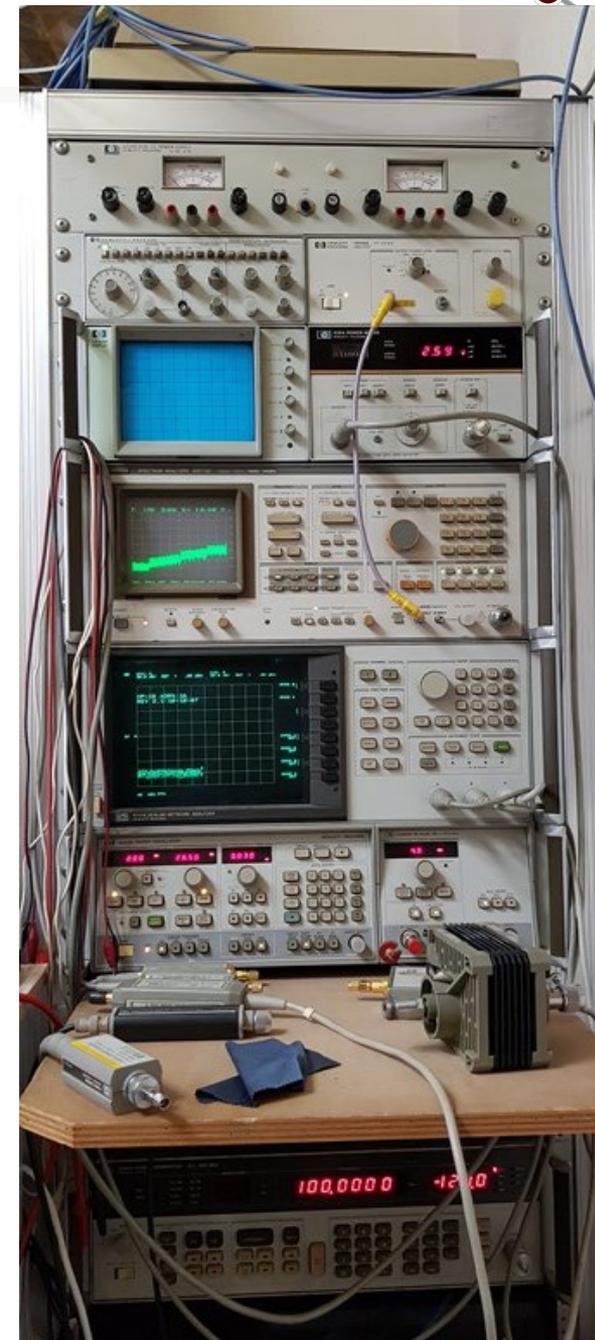
● RC5B-3 + ERC5A

● 1.6m パラボラ



外部公開計測環境

- HP 6253A Dual DC power supply
- HP 3312A Function Generator
- HP 11975A 2G–8G Amplifier for Spectrum Analyzer Local frequency
- HP 1340A X–Y monitor for HP 8970A noise figure measurement display
- HP 7470A X–Y Plotter for HP 8757A measurement result printing
- HP 436A Microwave Power Meter
- HP 8487A Power Sensor 50MHz–50GHz APC2.4
- Anritsu MS710F modified version as same as MS710E(External mixer ready to 140GHz)
- ANRITSU MH680B Tracking Generator
- HP 8757A Scalar Network Analyzer
- HP 8350B Microwave Seep Generator 2GHz–26.5GHz
- HP 11664E AC Detector 10MHz–26.5GHz
- HP 11664A AC Detector 10MHz–18.0GHz
- HP 85025B AC/DC Detector 10MHz–26.5GHz
- HP 8510C Vector Network Analyzer
- HP 8514A S–Parameter Test Set 500MHz–18GHz
- JDSU JD724C Vector Network Analyzer 5MHz–4GHz
- HP 8970A Automatic Noise Figure Meter
- HP 346C Noise Source 10MHz–26.5GHz
- HP 8586A Signal Generator 100KHz–990MHz
- HP 5351B Frequency Counter 10MHz–26.5GHz
- Tektronix TDS784D Oscilloscope 1GHz/4Gsps 4ch



- 量子コンピュータとは？
 - 量子コンピュータの構成と構造
 - 量子とマイクロ波の関係
 - 量子ビットを理解する
 - ゲート
- デモ
 - SDRの選び方と見本（実機）
 - 量子ファームウェアのデモ

●量子コンピュータの構成と構造

● 種類

● アニーリング型

- グラフ問題しか解けない(汎用性がない)

● ゲート型

- 汎用コンピュータとして利用できる真のコンピュータ

- 古典コンピュータより早いという証明は無い

- 一部のアルゴリズムのみ証明されている

- Shorのアルゴリズム(素因数分解)

- 量子超越性 \div (現在)60~70量子ビット

● 種類

- 超電導型

- 実用化されている唯一のゲート型量子コンピュータ

- 光制御型

- 研究室レベル

- 半導体型

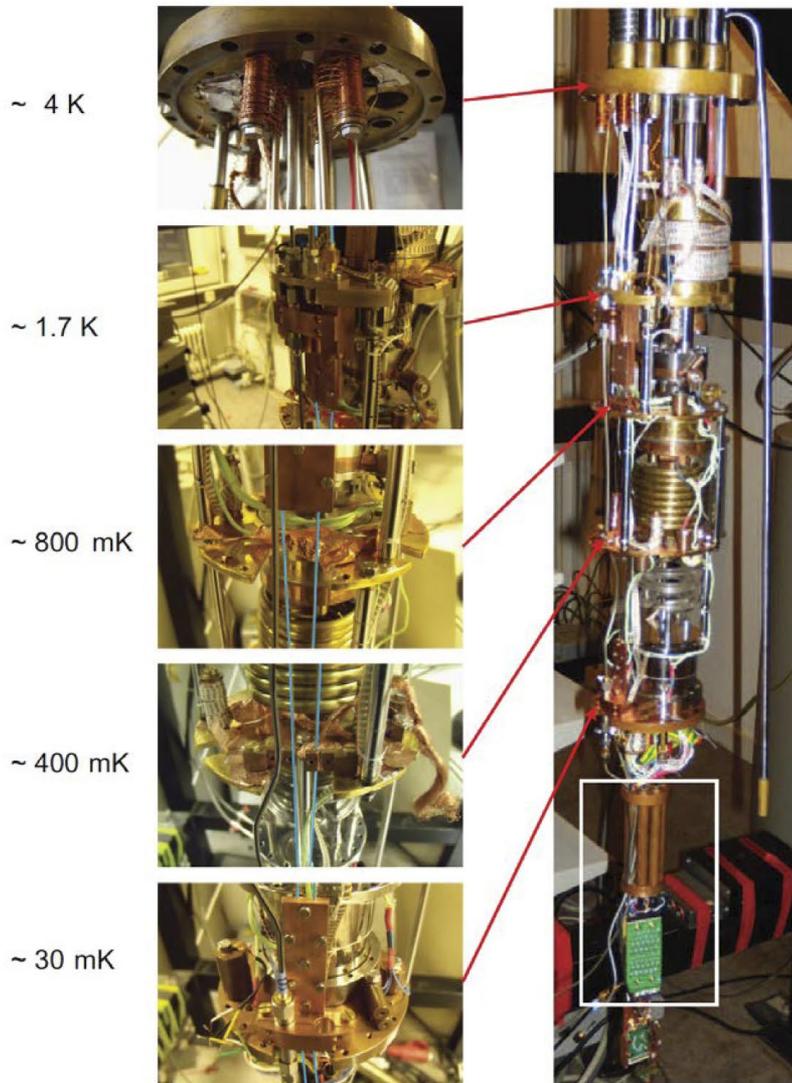
- 理論レベル

超電導量子コンピュータ

● IBM Q

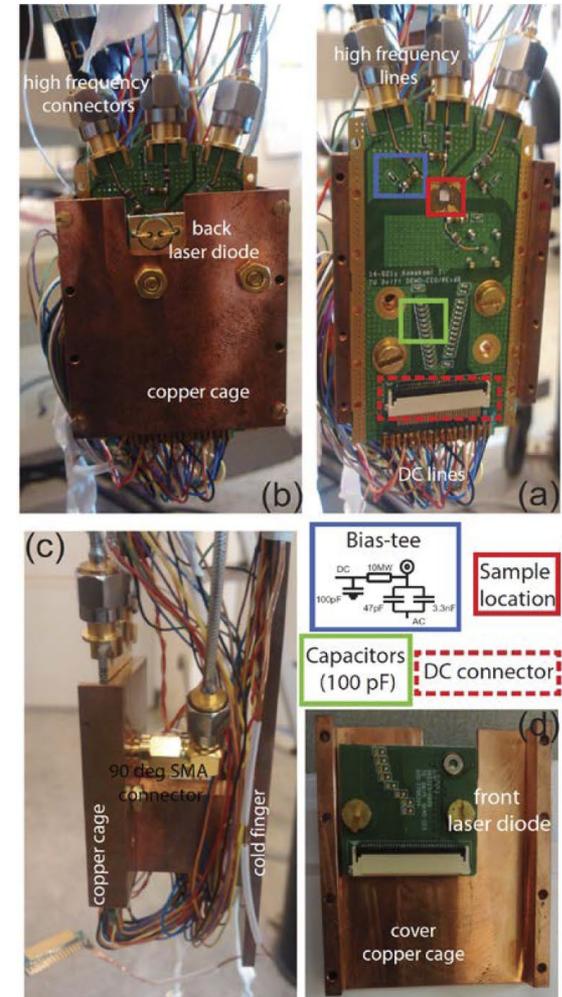


超電導量子コンピュータの温度



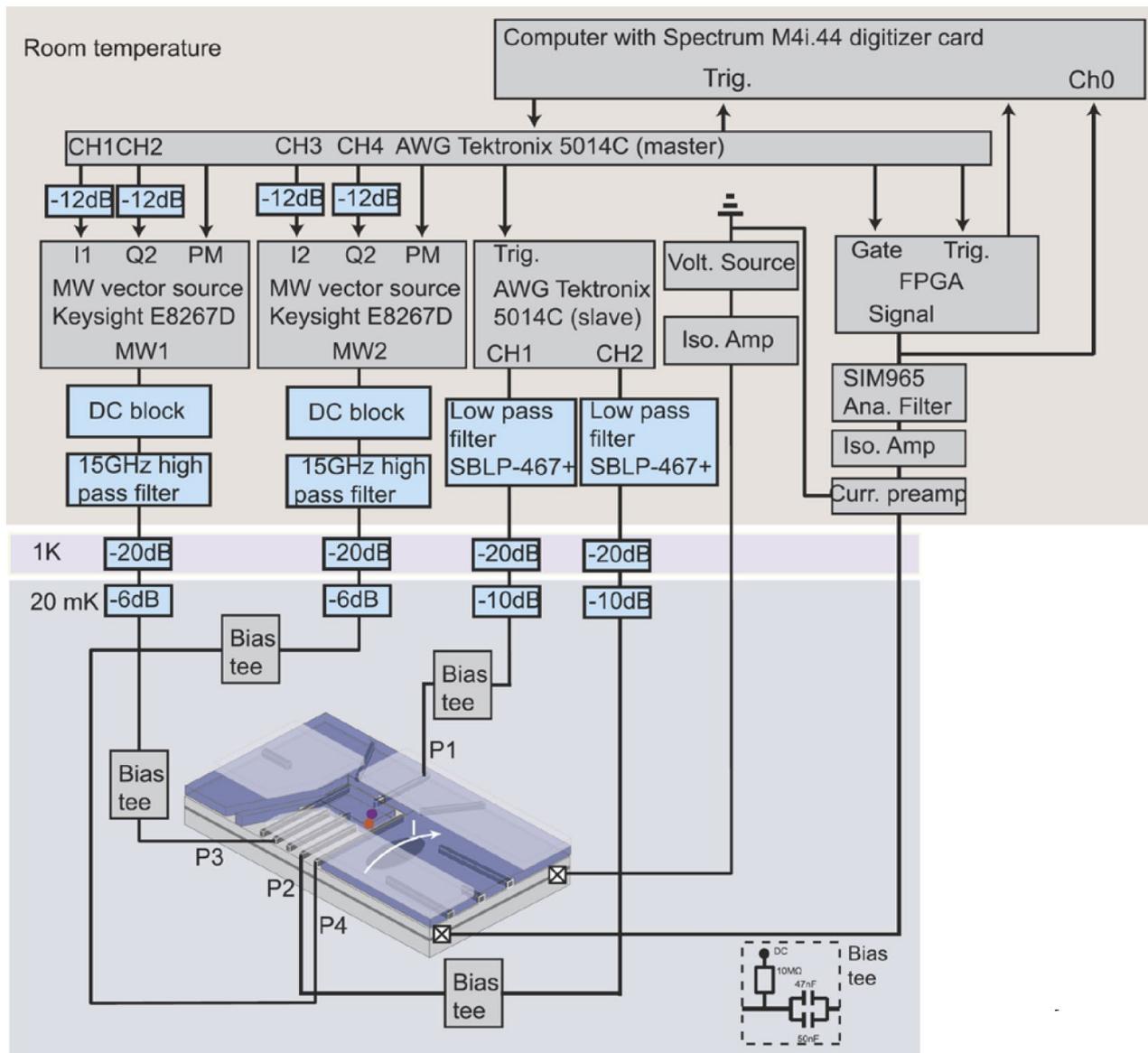
copper powder filters

RC filters

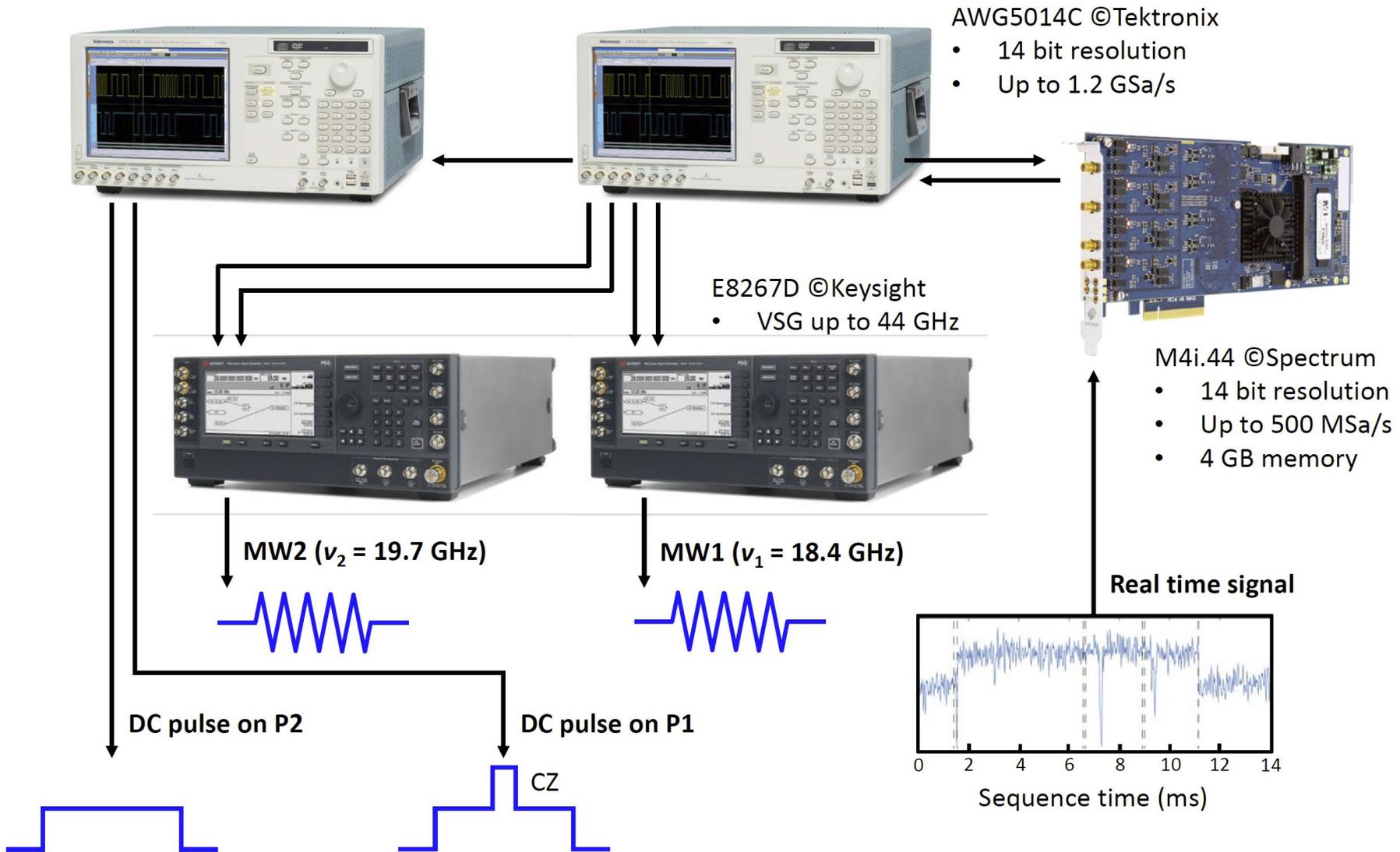


PhD dissertation, TU Delft (2016) Scarlino

超電導量子コンピュータの構成



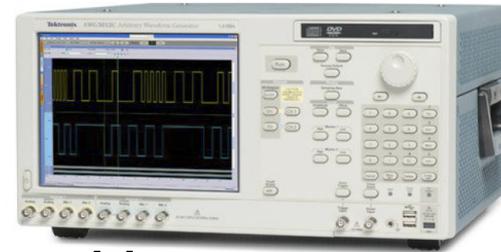
超電導量子コンピュータの構成



超電導量子コンピュータの構成

AWG5014C ©Tektronix

- 14 bit resolution
- Up to 1.2 GSa/s



E8267D ©Keysight

- VSG up to 44 GHz

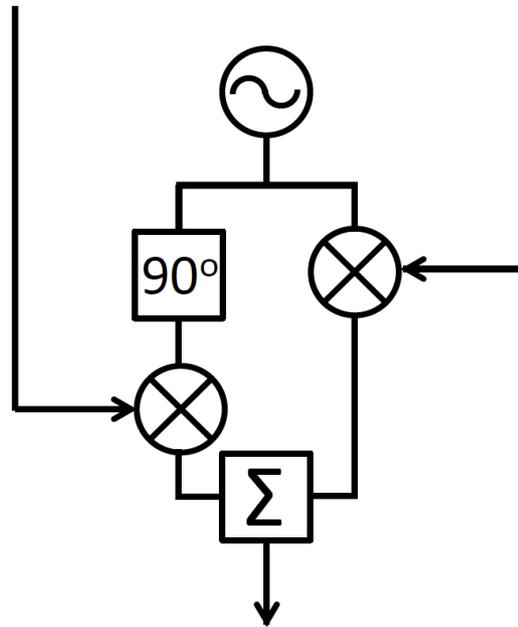


MW1 ($\nu_1 = 18.4$ GHz)



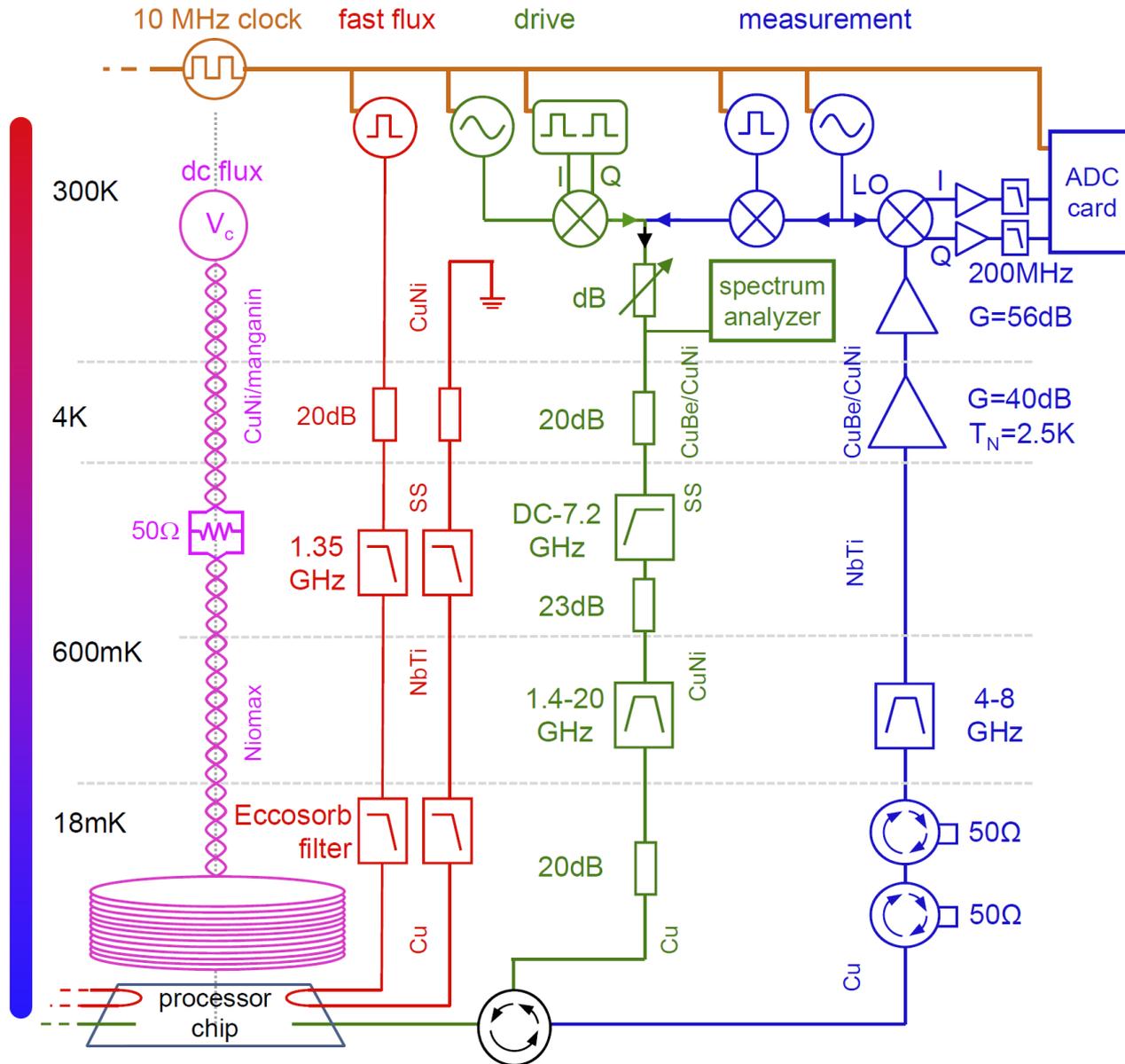
$$I(t) = A(t) \cos(2\pi\nu_m t + \theta)$$

$$Q(t) = A(t) \sin(2\pi\nu_m t + \theta)$$

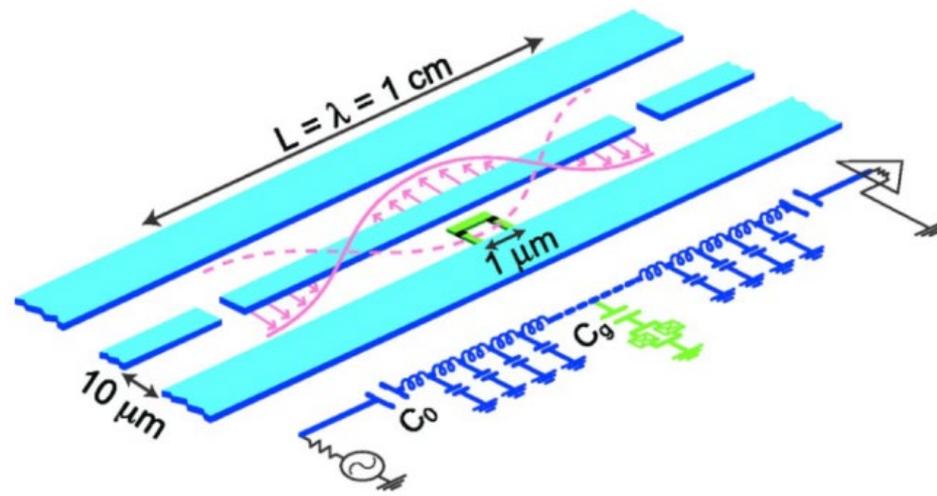
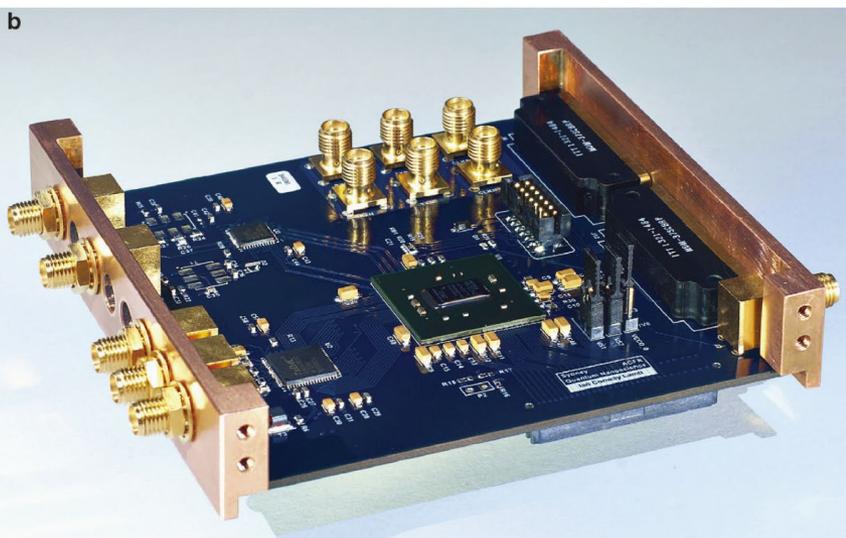
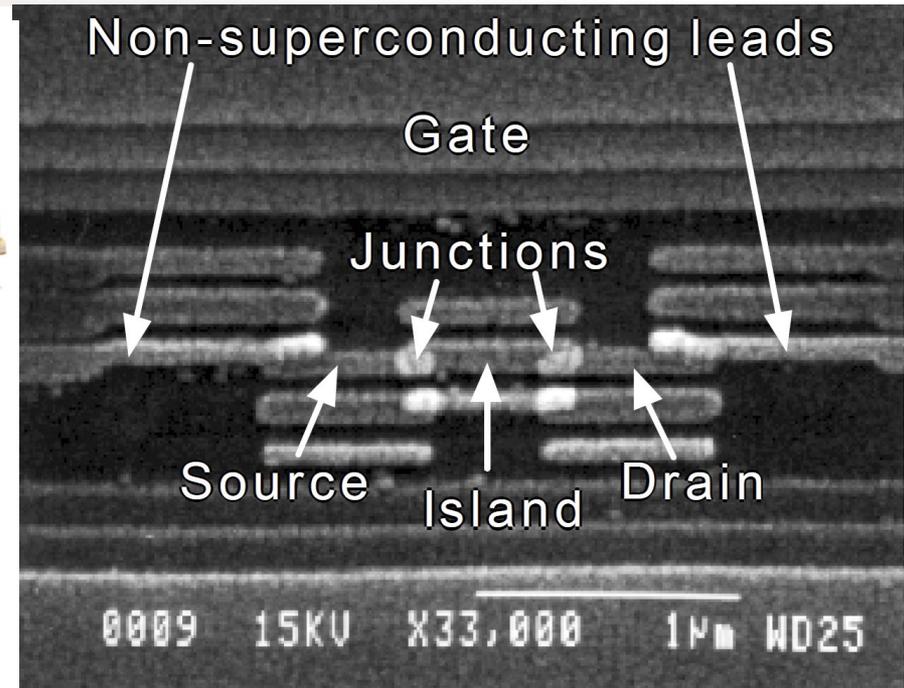
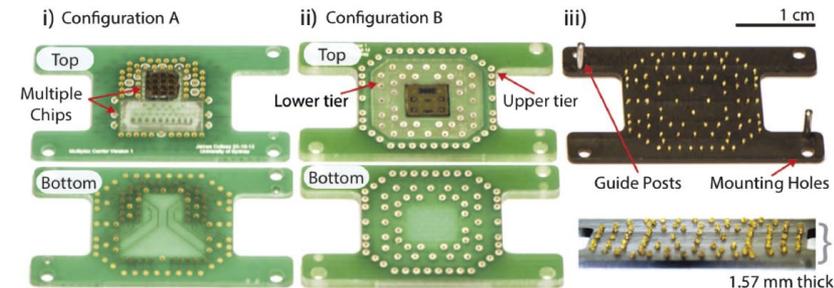
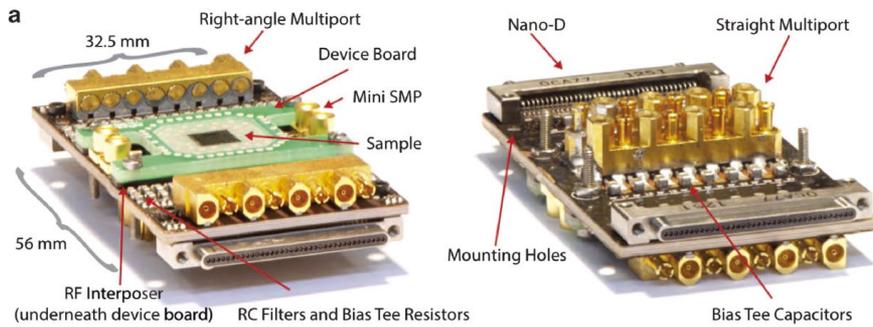


$$I(t) \cos(2\pi\nu_{LO} t) - Q(t) \sin(2\pi\nu_{LO} t) = A(t) \cos[2\pi(\nu_{LO} + \nu_m)t + \theta]$$

超電導量子コンピュータの構成



超電導量子コンピュータの構成



●量子とマイクロ波の関係

波動関数の変形

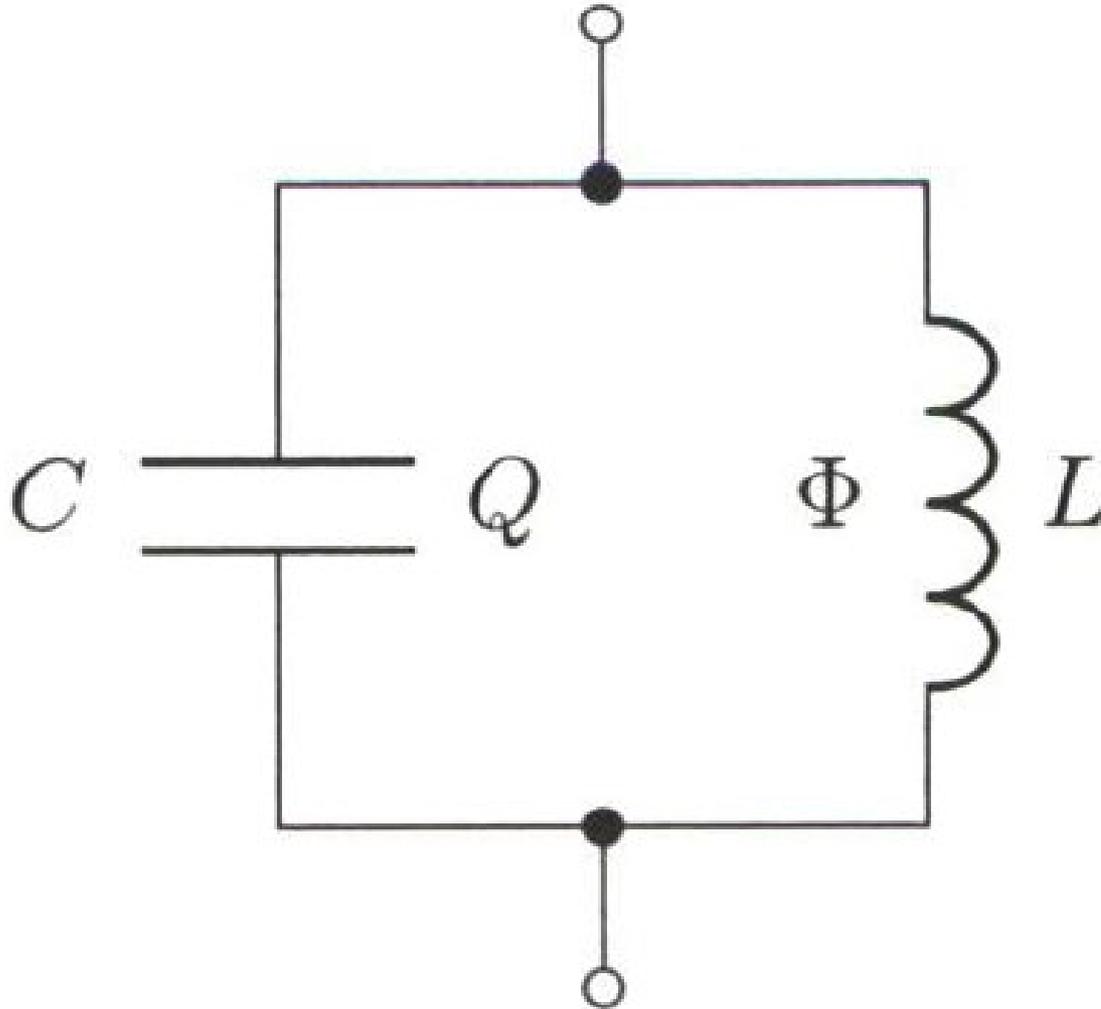
- 時間に依存するシュレーディンガー方程式によって記述された量子系の状態ベクトルおよび波動関数の時間的变化

$$i\hbar \frac{d}{dt} |\psi(t)\rangle = \hat{H} |\psi(t)\rangle .$$

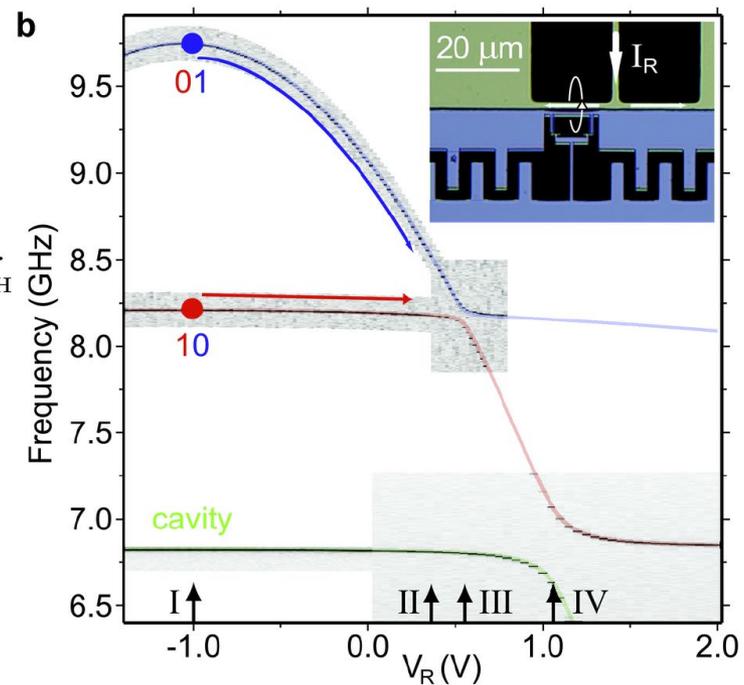
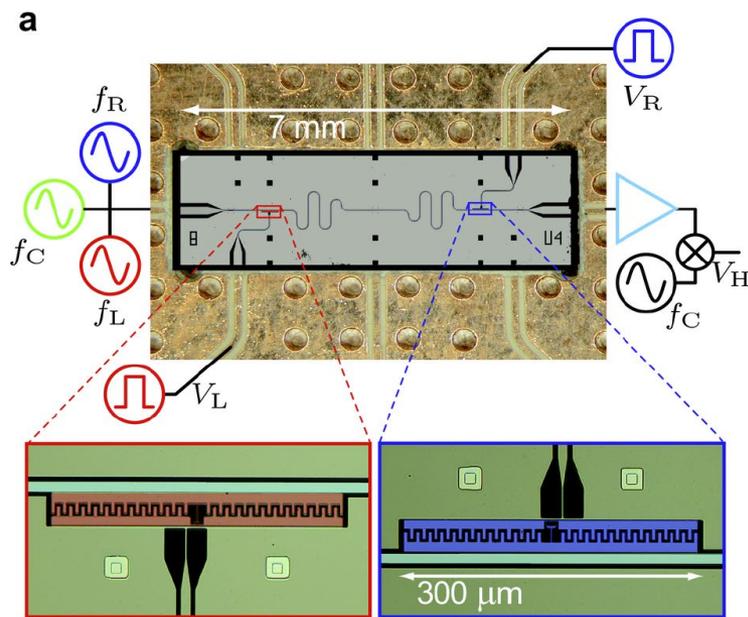
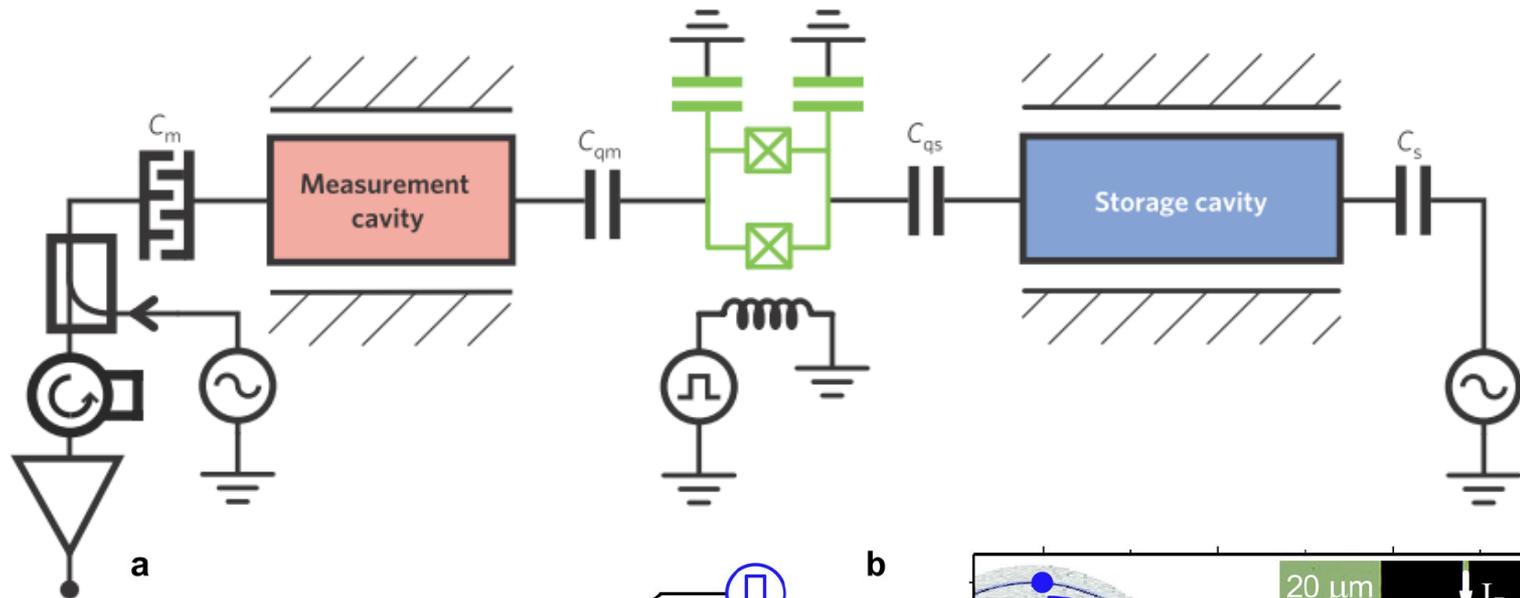
- 古典的な電磁場と相互作用する電子のハミルトニアン

$$\hat{H} = \frac{1}{2m} (\hat{\mathbf{p}} + e\mathbf{A}(\hat{\mathbf{x}}, t))^2 - e\Phi(\hat{\mathbf{x}}, t) .$$

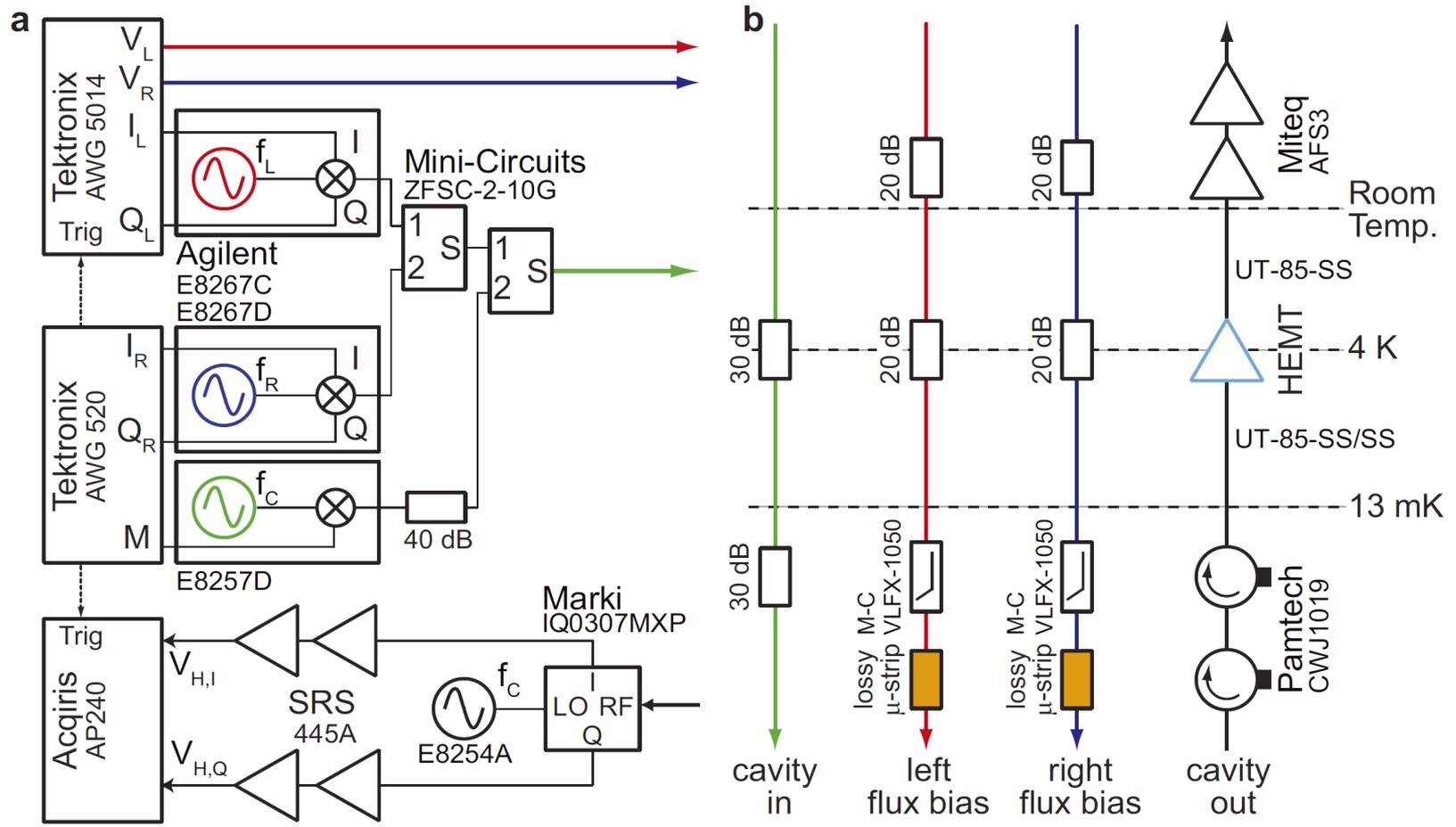
波動関数の変形



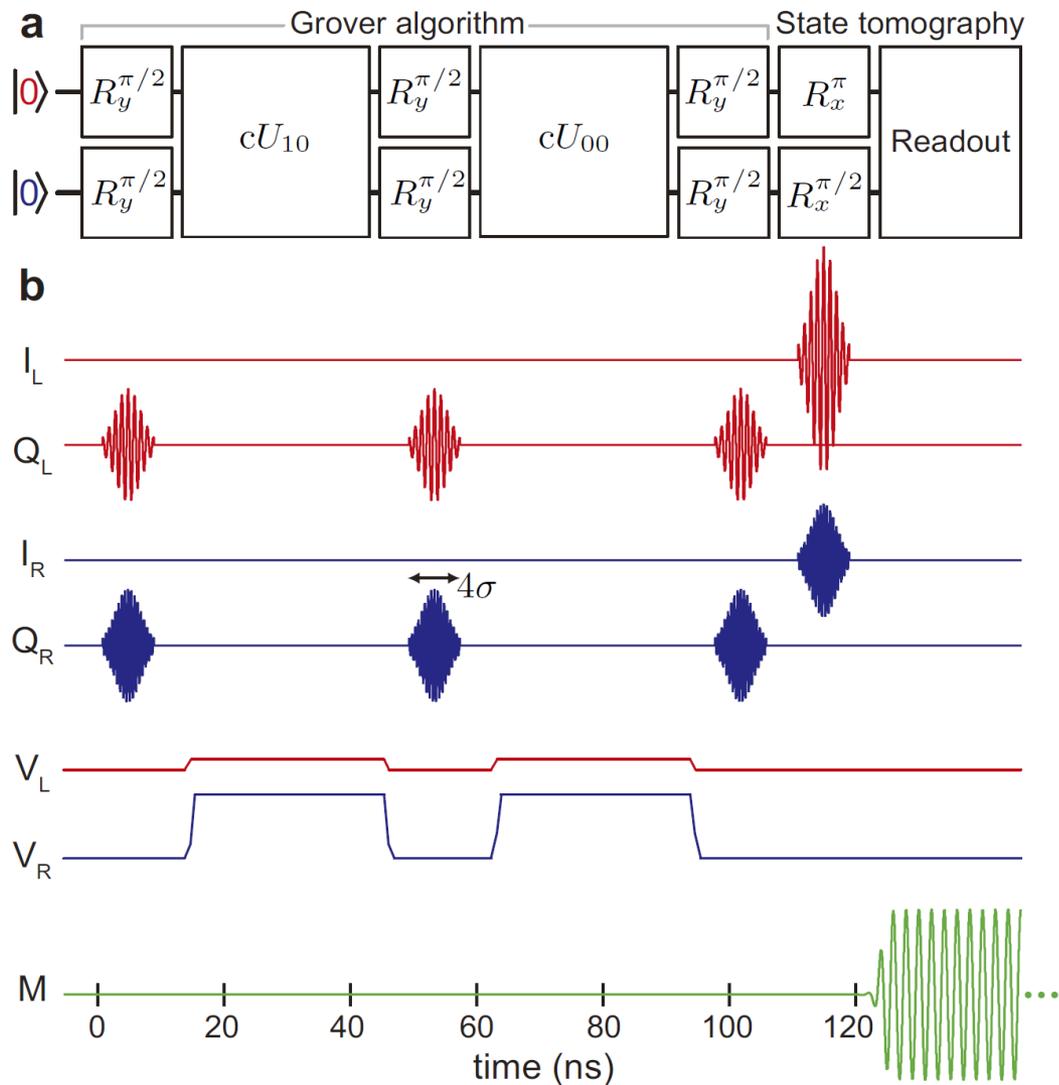
トランズモン量子ビット: 1量子ビット



トランズモン1量子ビットの操作構成

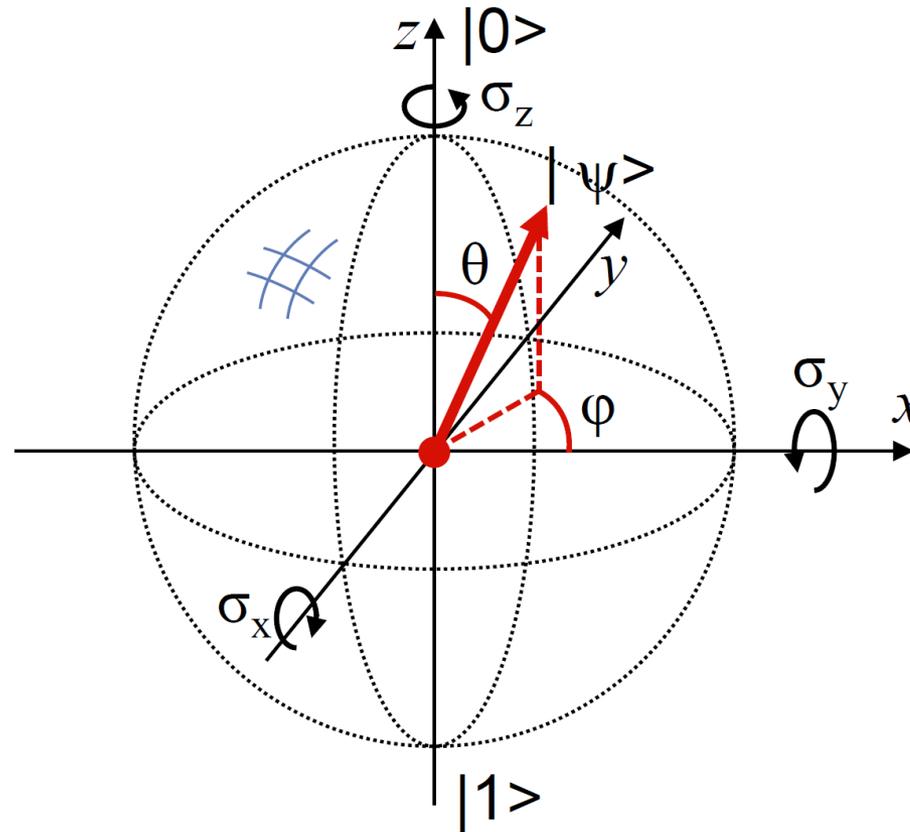


トランズモン1量子ビットの操作内容

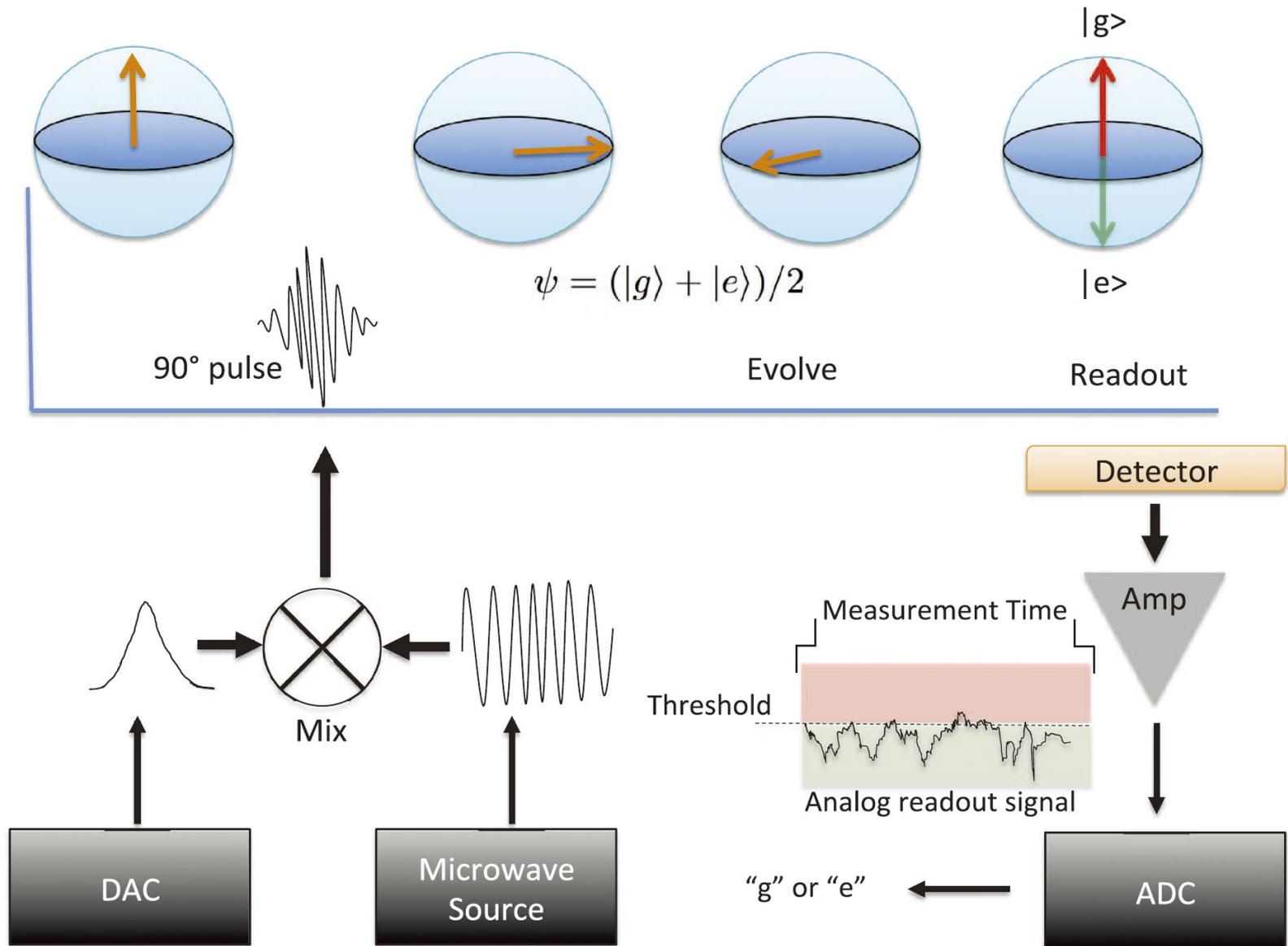


●量子ビットの状態

- 量子ビットの状態を表記する方法の一つ

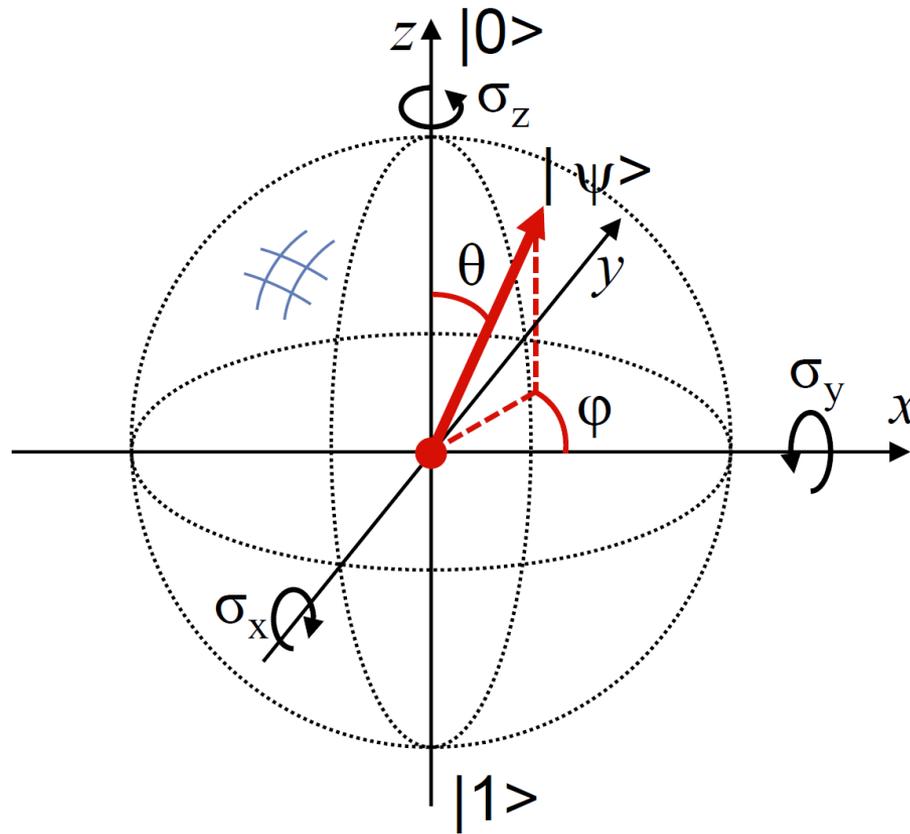


量子ビットの操作

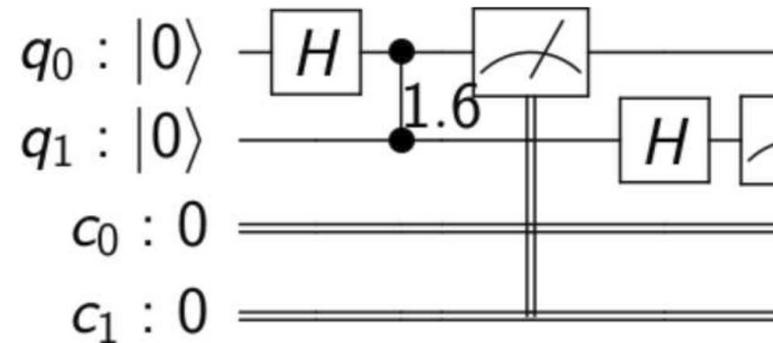
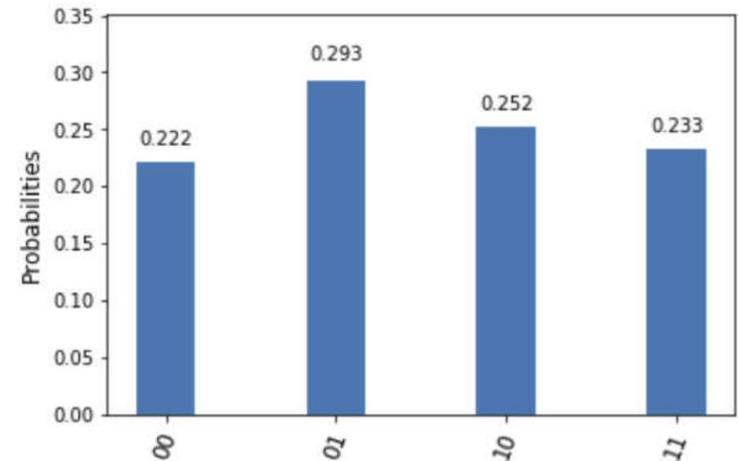


ブロッホ球とビット表記

- 量子は確率で存在するため、アナログ量となる

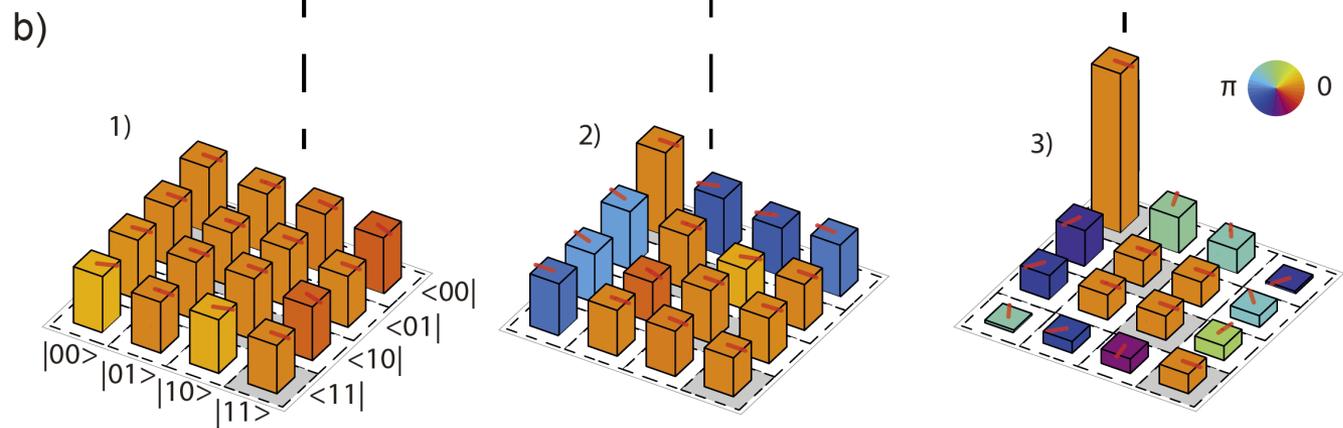
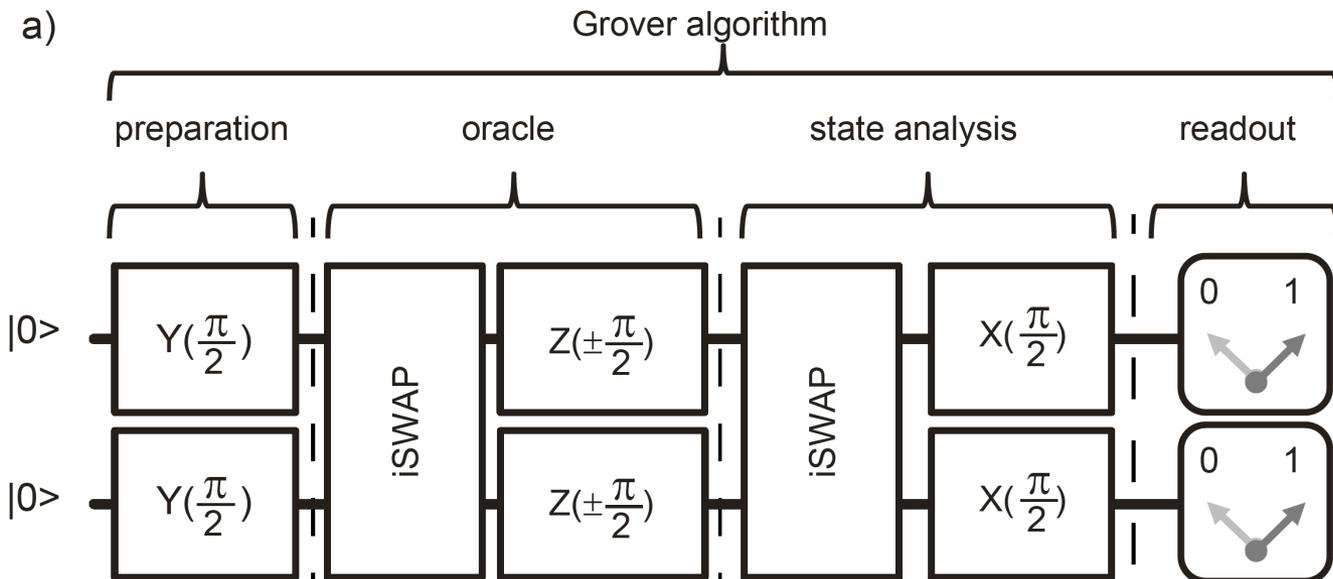


{'00': 227, '01': 300, '10': 258, '11': 239}



3次元のコンスタレーション表記

● もう一つの量子ビットの状態を表記する方法

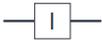
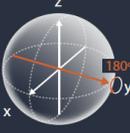
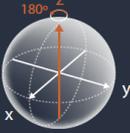
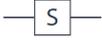
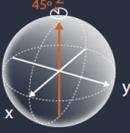
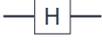


●量子ビットの操作単位

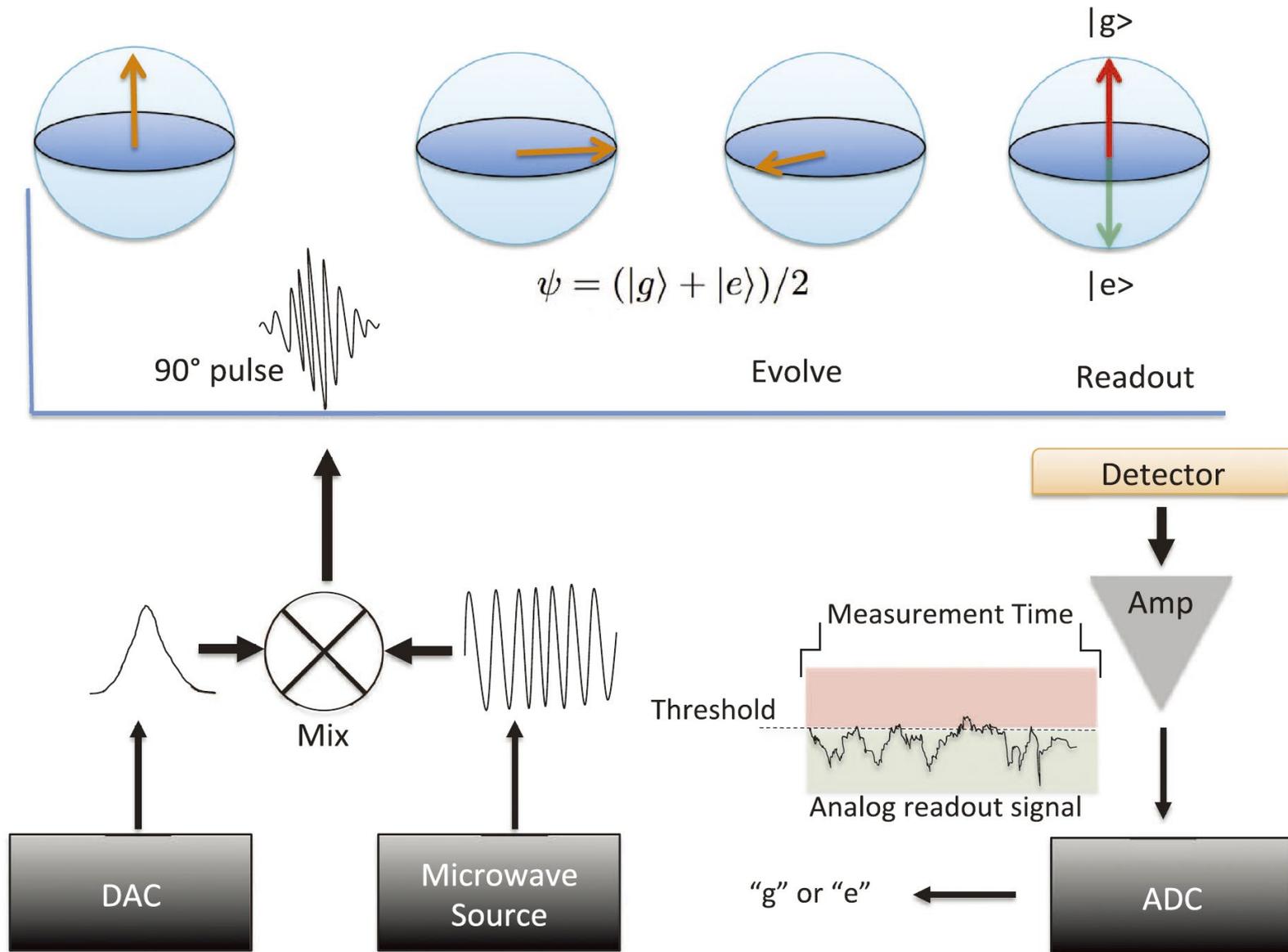
古典コンピュータの操作単位

GATE		CIRCUIT SYMBOL	TRUTH TABLE										
NOT	The output is 1 when the input is 0 and 0 when the input is 1.		<table border="1"><thead><tr><th>Input</th><th>Output</th></tr></thead><tbody><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></tbody></table>	Input	Output	0	1	1	0				
Input	Output												
0	1												
1	0												
AND	The output is 1 only when both inputs are 1, otherwise the output is 0.		<table border="1"><thead><tr><th>Input</th><th>Output</th></tr></thead><tbody><tr><td>0 0</td><td>0</td></tr><tr><td>0 1</td><td>0</td></tr><tr><td>1 0</td><td>0</td></tr><tr><td>1 1</td><td>1</td></tr></tbody></table>	Input	Output	0 0	0	0 1	0	1 0	0	1 1	1
Input	Output												
0 0	0												
0 1	0												
1 0	0												
1 1	1												
OR	The output is 0 only when both inputs are 0, otherwise the output is 1.		<table border="1"><thead><tr><th>Input</th><th>Output</th></tr></thead><tbody><tr><td>0 0</td><td>0</td></tr><tr><td>0 1</td><td>1</td></tr><tr><td>1 0</td><td>1</td></tr><tr><td>1 1</td><td>1</td></tr></tbody></table>	Input	Output	0 0	0	0 1	1	1 0	1	1 1	1
Input	Output												
0 0	0												
0 1	1												
1 0	1												
1 1	1												
NAND	The output is 0 only when both inputs are 1, otherwise the output is 1.		<table border="1"><thead><tr><th>Input</th><th>Output</th></tr></thead><tbody><tr><td>0 0</td><td>1</td></tr><tr><td>0 1</td><td>1</td></tr><tr><td>1 0</td><td>1</td></tr><tr><td>1 1</td><td>0</td></tr></tbody></table>	Input	Output	0 0	1	0 1	1	1 0	1	1 1	0
Input	Output												
0 0	1												
0 1	1												
1 0	1												
1 1	0												
NOR	The output is 1 only when both inputs are 0, otherwise the output is 0.		<table border="1"><thead><tr><th>Input</th><th>Output</th></tr></thead><tbody><tr><td>0 0</td><td>1</td></tr><tr><td>0 1</td><td>0</td></tr><tr><td>1 0</td><td>0</td></tr><tr><td>1 1</td><td>0</td></tr></tbody></table>	Input	Output	0 0	1	0 1	0	1 0	0	1 1	0
Input	Output												
0 0	1												
0 1	0												
1 0	0												
1 1	0												
XOR	The output is 1 only when the two inputs have different value, otherwise the output is 0.		<table border="1"><thead><tr><th>Input</th><th>Output</th></tr></thead><tbody><tr><td>0 0</td><td>0</td></tr><tr><td>0 1</td><td>1</td></tr><tr><td>1 0</td><td>1</td></tr><tr><td>1 1</td><td>0</td></tr></tbody></table>	Input	Output	0 0	0	0 1	1	1 0	1	1 1	0
Input	Output												
0 0	0												
0 1	1												
1 0	1												
1 1	0												
XNOR	The output is 1 only when the two inputs have the same value, otherwise the output is 0.		<table border="1"><thead><tr><th>Input</th><th>Output</th></tr></thead><tbody><tr><td>0 0</td><td>1</td></tr><tr><td>0 1</td><td>0</td></tr><tr><td>1 0</td><td>0</td></tr><tr><td>1 1</td><td>1</td></tr></tbody></table>	Input	Output	0 0	1	0 1	0	1 0	0	1 1	1
Input	Output												
0 0	1												
0 1	0												
1 0	0												
1 1	1												

量子コンピュータの操作単位

GATE	CIRCUIT REPRESENTATION	MATRIX REPRESENTATION	TRUTH TABLE	BLOCH SPHERE						
I Identity-gate: no rotation is performed.		$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	<table border="1"> <thead> <tr> <th>Input</th> <th>Output</th> </tr> </thead> <tbody> <tr> <td> 0⟩</td> <td> 0⟩</td> </tr> <tr> <td> 1⟩</td> <td> 1⟩</td> </tr> </tbody> </table>	Input	Output	0⟩	0⟩	1⟩	1⟩	
Input	Output									
0⟩	0⟩									
1⟩	1⟩									
X gate: rotates the qubit state by π radians (180°) about the x-axis.		$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	<table border="1"> <thead> <tr> <th>Input</th> <th>Output</th> </tr> </thead> <tbody> <tr> <td> 0⟩</td> <td> 1⟩</td> </tr> <tr> <td> 1⟩</td> <td> 0⟩</td> </tr> </tbody> </table>	Input	Output	0⟩	1⟩	1⟩	0⟩	
Input	Output									
0⟩	1⟩									
1⟩	0⟩									
Y gate: rotates the qubit state by π radians (180°) about the y-axis.		$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	<table border="1"> <thead> <tr> <th>Input</th> <th>Output</th> </tr> </thead> <tbody> <tr> <td> 0⟩</td> <td>i 1⟩</td> </tr> <tr> <td> 1⟩</td> <td>-i 0⟩</td> </tr> </tbody> </table>	Input	Output	0⟩	i 1⟩	1⟩	-i 0⟩	
Input	Output									
0⟩	i 1⟩									
1⟩	-i 0⟩									
Z gate: rotates the qubit state by π radians (180°) about the z-axis.		$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	<table border="1"> <thead> <tr> <th>Input</th> <th>Output</th> </tr> </thead> <tbody> <tr> <td> 0⟩</td> <td> 0⟩</td> </tr> <tr> <td> 1⟩</td> <td>- 1⟩</td> </tr> </tbody> </table>	Input	Output	0⟩	0⟩	1⟩	- 1⟩	
Input	Output									
0⟩	0⟩									
1⟩	- 1⟩									
S gate: rotates the qubit state by $\frac{\pi}{2}$ radians (90°) about the z-axis.		$S = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{pmatrix}$	<table border="1"> <thead> <tr> <th>Input</th> <th>Output</th> </tr> </thead> <tbody> <tr> <td> 0⟩</td> <td> 0⟩</td> </tr> <tr> <td> 1⟩</td> <td>$e^{i\frac{\pi}{2}} 1\rangle$</td> </tr> </tbody> </table>	Input	Output	0⟩	0⟩	1⟩	$e^{i\frac{\pi}{2}} 1\rangle$	
Input	Output									
0⟩	0⟩									
1⟩	$e^{i\frac{\pi}{2}} 1\rangle$									
T gate: rotates the qubit state by $\frac{\pi}{4}$ radians (45°) about the z-axis.		$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}$	<table border="1"> <thead> <tr> <th>Input</th> <th>Output</th> </tr> </thead> <tbody> <tr> <td> 0⟩</td> <td> 0⟩</td> </tr> <tr> <td> 1⟩</td> <td>$e^{i\frac{\pi}{4}} 1\rangle$</td> </tr> </tbody> </table>	Input	Output	0⟩	0⟩	1⟩	$e^{i\frac{\pi}{4}} 1\rangle$	
Input	Output									
0⟩	0⟩									
1⟩	$e^{i\frac{\pi}{4}} 1\rangle$									
H gate: rotates the qubit state by π radians (180°) about an axis diagonal in the x-z plane. This is equivalent to an X-gate followed by a $\frac{\pi}{2}$ rotation about the y-axis.		$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$	<table border="1"> <thead> <tr> <th>Input</th> <th>Output</th> </tr> </thead> <tbody> <tr> <td> 0⟩</td> <td>$\frac{ 0\rangle + 1\rangle}{\sqrt{2}}$</td> </tr> <tr> <td> 1⟩</td> <td>$\frac{ 0\rangle - 1\rangle}{\sqrt{2}}$</td> </tr> </tbody> </table>	Input	Output	0⟩	$\frac{ 0\rangle + 1\rangle}{\sqrt{2}}$	1⟩	$\frac{ 0\rangle - 1\rangle}{\sqrt{2}}$	
Input	Output									
0⟩	$\frac{ 0\rangle + 1\rangle}{\sqrt{2}}$									
1⟩	$\frac{ 0\rangle - 1\rangle}{\sqrt{2}}$									

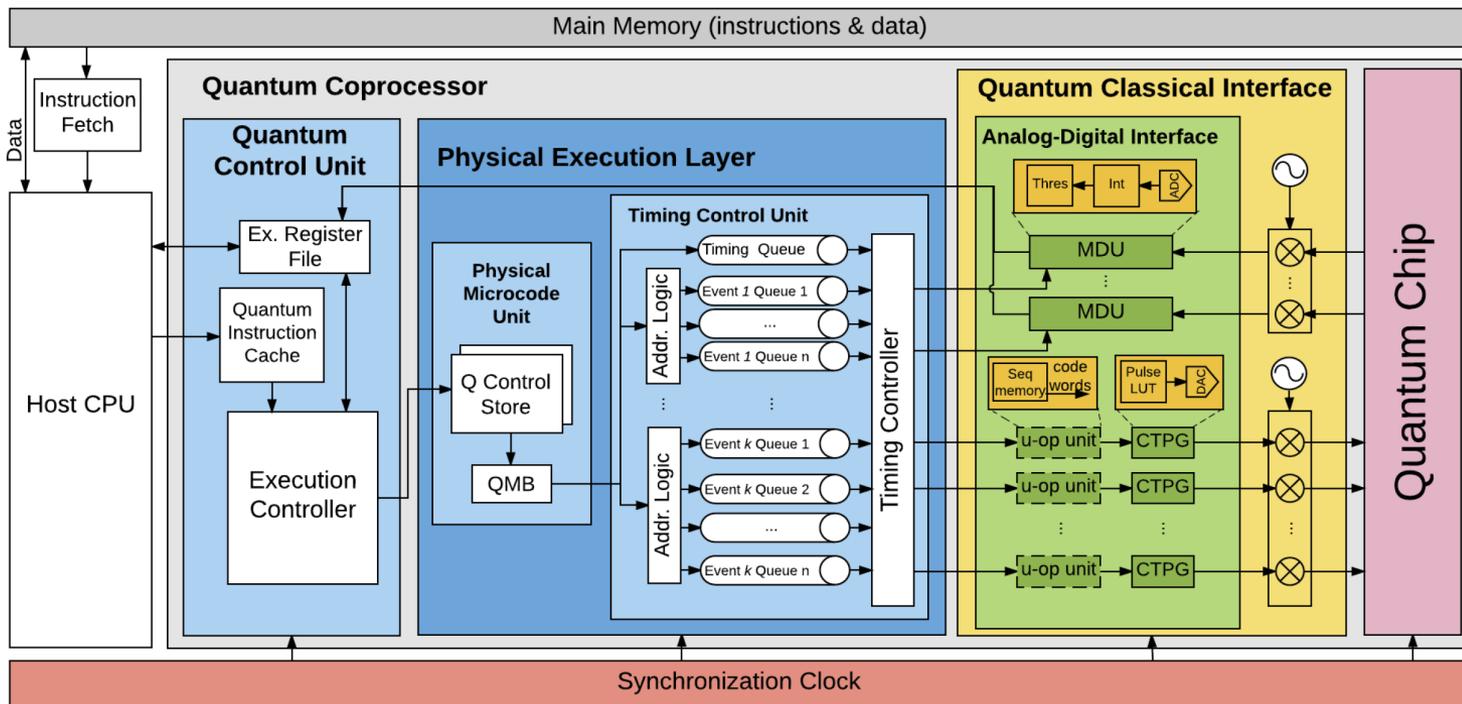
量子ビットの操作



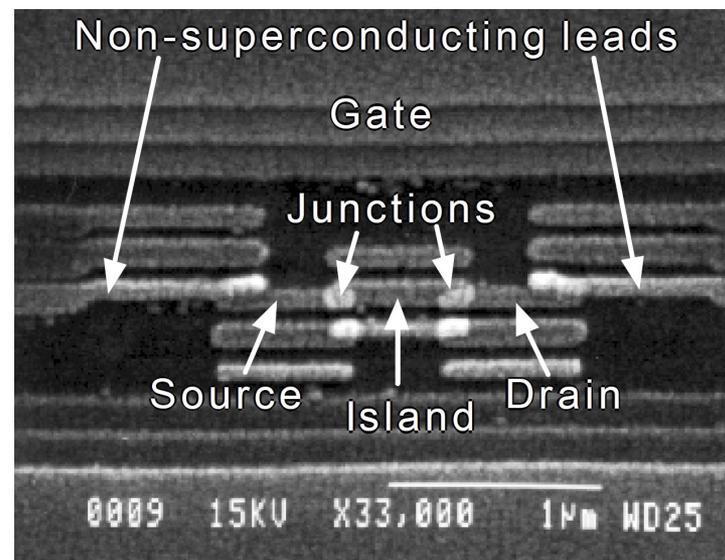
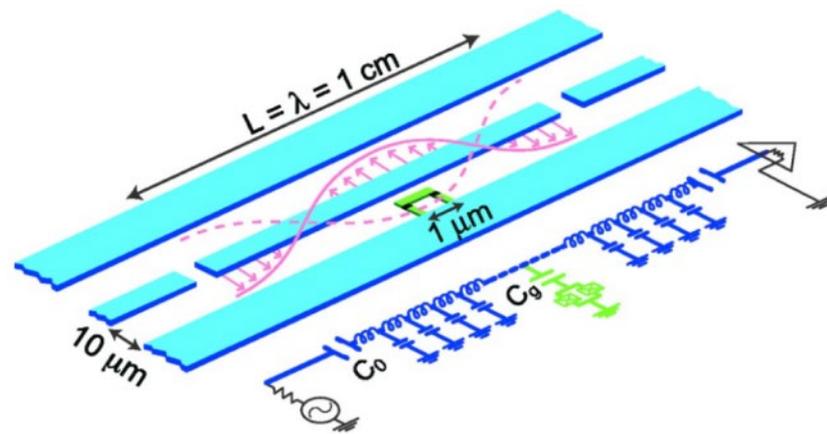
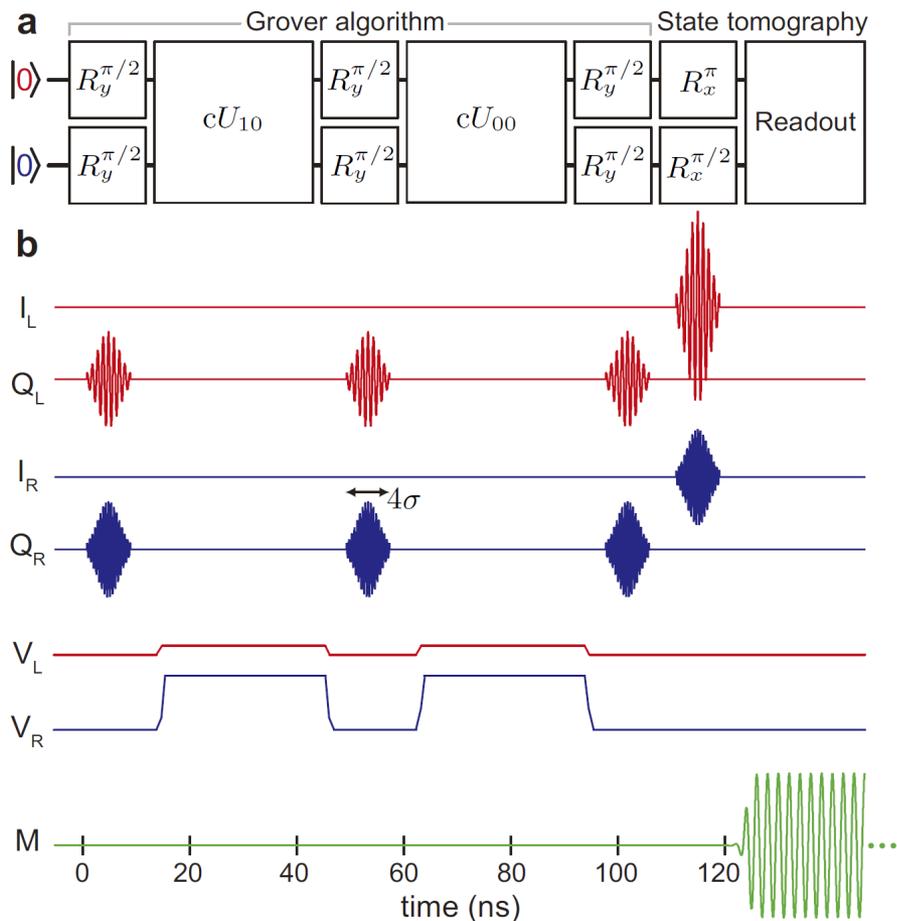
●量子ファームウェア

量子ファームウェアとは？

- 量子ビットを物理的に操作するもの
 - 量子ゲートに合わせた波形を生成する
 - 複数量子ビットの場合は同期をとる
 - 量子ビットへの高周波の入出力
 - 取得した量子の状態からビットへ変換する



量子ファームウェアの実体



- 量子ファームウェアとGNURadio

SDR用のデジタル送受信ソフトウェア

● GNURadio

● オープンソースのソフトウェア信号処理器

● フロー・ダイアグラムによる実装が可能

● デジタル処理を行うためのソフトウェア

● 同様の機能を持つ商用のソフトウェア信号処理器

- MATLABのSimulink

- LabVIEWのVI

● 動作環境

● Windows and Linux

- <https://www.gnuradio.org/>

● GRC(GNURadio-companion)

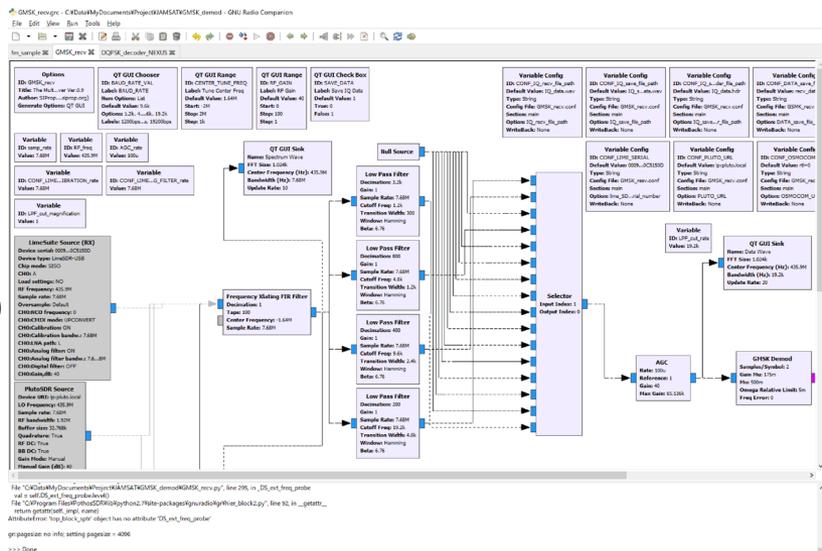
● フローダイアグラムエディタ

● ブロック

- 信号処理を行う単位

● フロー

- 信号の伝達ライン



SDRとはなにか？

● Software Defined Radio

=ソフトウェアで定義された無線機

● 一番簡略化した構造

● 構成物

● アンテナ

● ADC/DAC

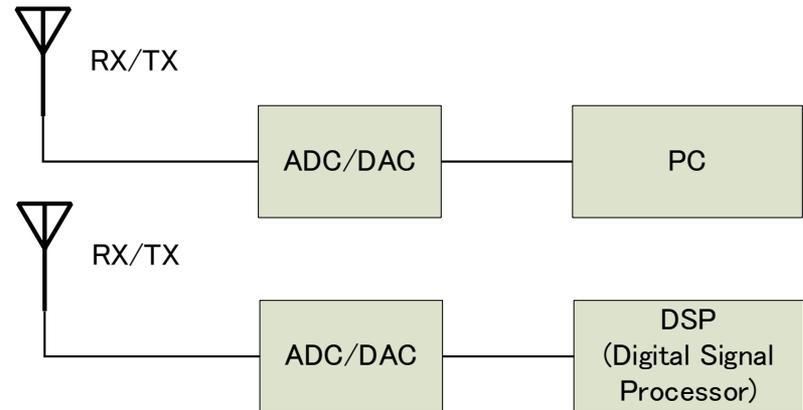
● 処理器(DSP=Digital Signal Processor)

● ソフトウェアで処理を定義できる信号処理に特化したプロセッサ

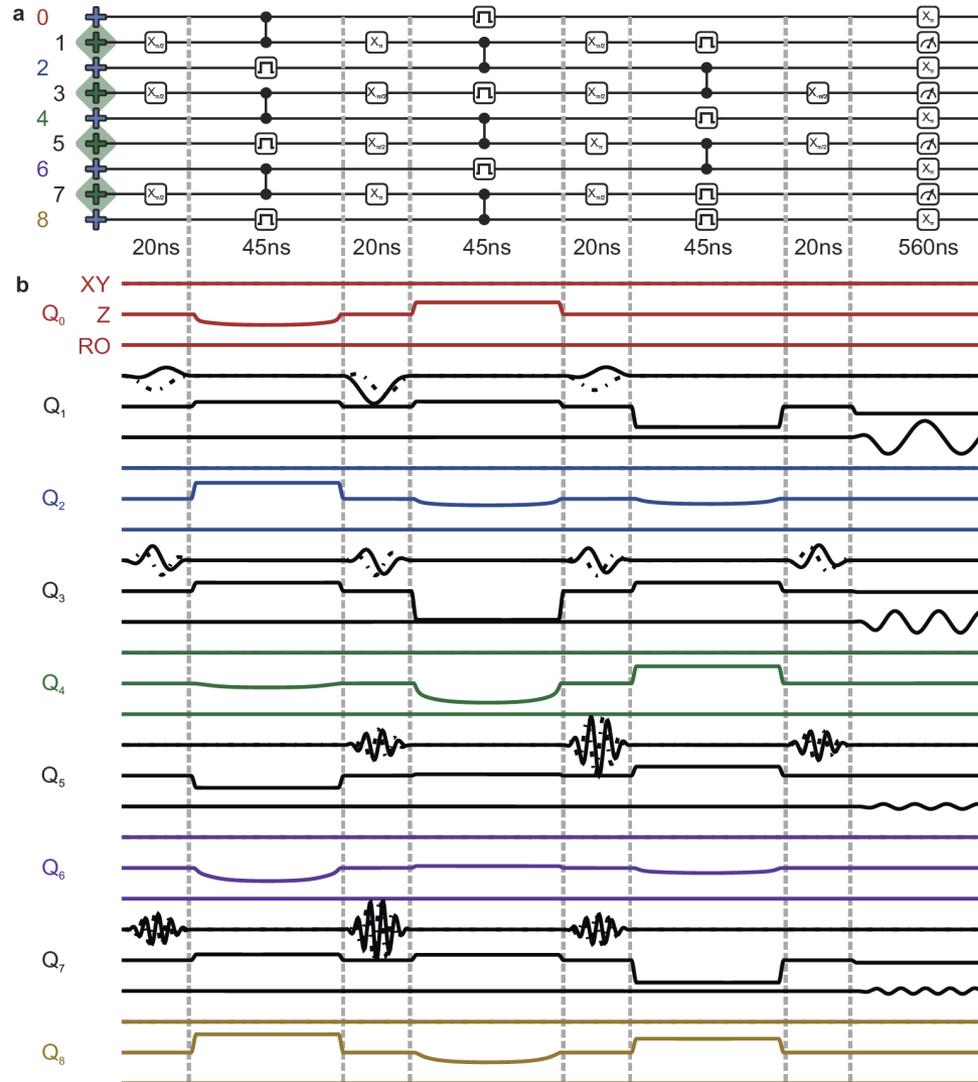
● PCで代用可能！

● 回路で処理しているものをDSPで処理しているだけ！

● 電波をDSPで処理できる形にする変換器≒SDR



多量子ビットの操作表記



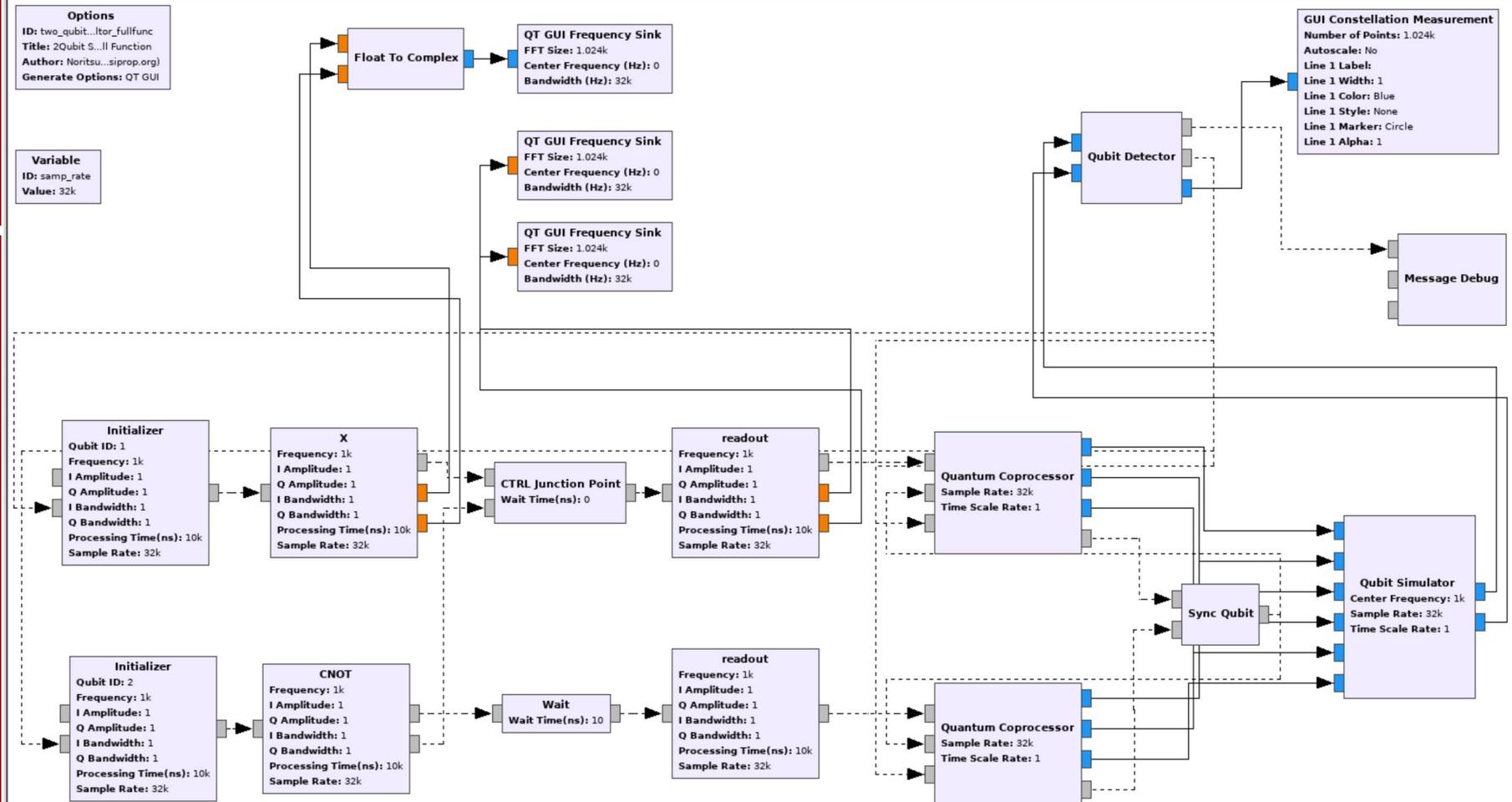
GNURadioの量子ゲート拡張

2qubit_simulator_fullfunc.grc - /home/noritsuna/gr-quantum/gr-quantum/examples - GNU Radio Companion

File Edit View Run Tools Help



1qubit_simulator_simple x 1qubit_simulator_compiler x 1qubit_simulator_SDR x 2qubit_simulator_simple x 2qubit_simulator_fullfunc x



量子ゲートと量子操作ブロック

- ▶ (no module specified)
- ▶ CCSDS
- ▶ Core
- ▼ Quantum
 - ▼ controllers
 - Initializer
 - Quantum Coprocessor
 - Qubit Simulator
 - Sync Qubit
 - Wait
 - readout
 - ▼ gates
 - CNOT
 - CTRL Junction Point
 - H
 - S
 - T
 - X
 - Y
 - Z
 - ▼ measurements
 - GUI Bloch Measurement
 - GUI Constellation Measurement
 - Qubit Detector
- ▶ Satellites
- ▶ habets



1量子ビットをシミュレータで操作する

SIPProp

1qubit_simulator_simple.grc - /home/noritsuna/gr-quantum/gr-quantum/examples - GNU Radio Companion

File Edit View Run Tools Help



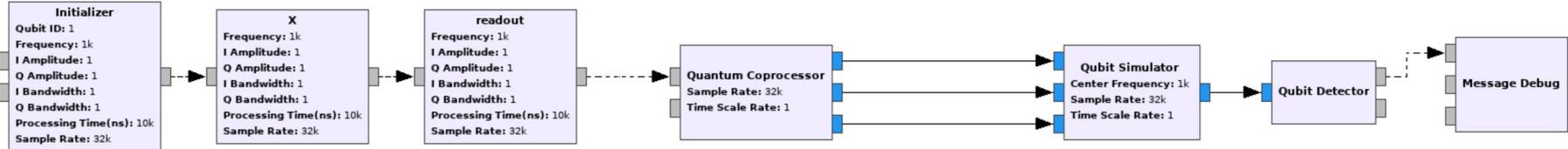
1qubit_simulator_simple 1qubit_simulator_compiler 1qubit_simulator_SDR 2qubit_simulator_simple 2qubit_simulator_fullfunc

Options

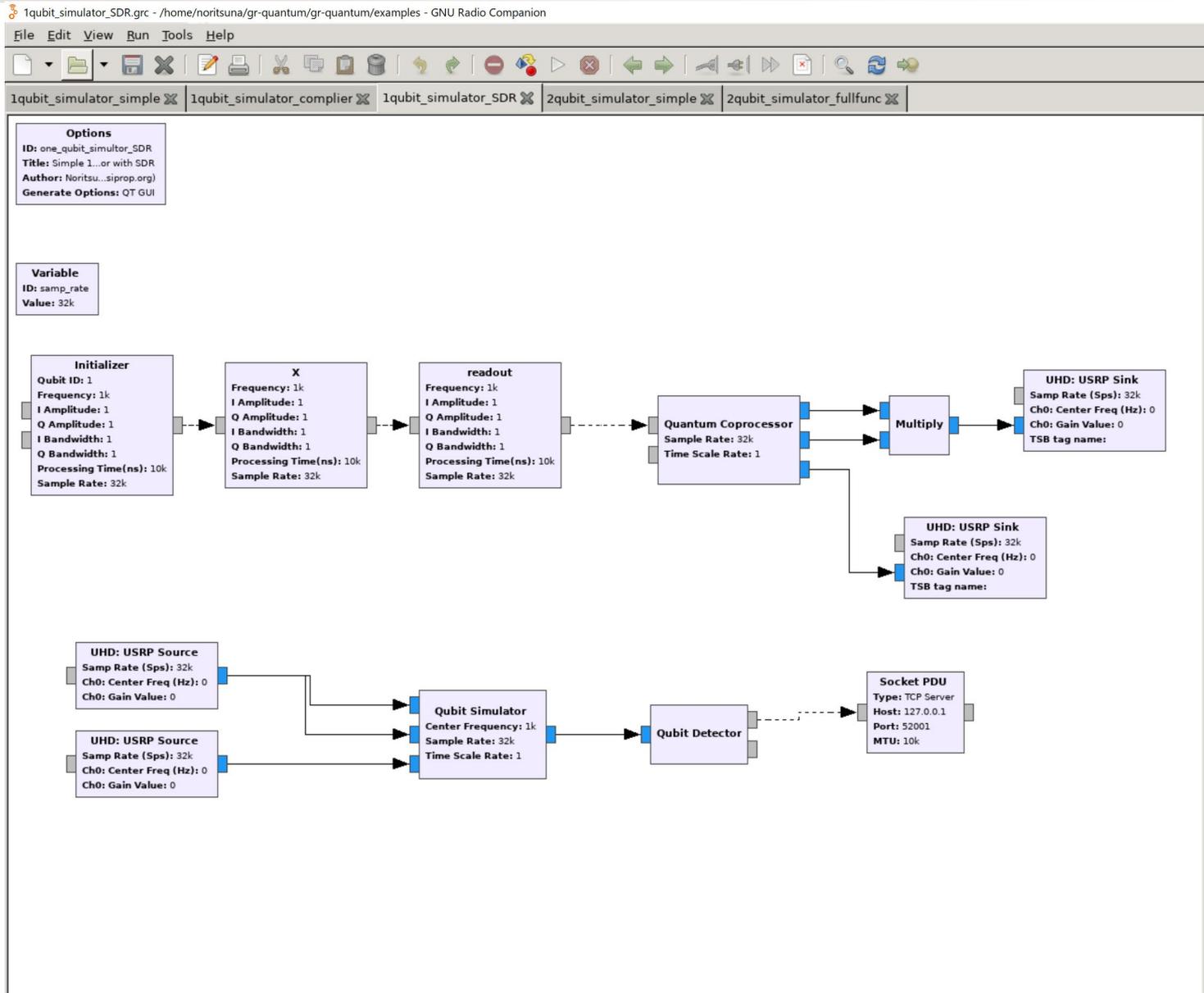
ID: one_qubit_simulator_simple
 Title: Simple 1Qubit Simulator
 Author: Noritsu...siprop.org)
 Generate Options: QT GUI

Variable

ID: samp_rate
 Value: 32k



1量子ビットをSDRで操作する



2量子ビットをシミュレータで操作する



2qubit_simulator_simple.grc - /home/noritsu/gr-quantum/gr-quantum/examples - GNU Radio Companion

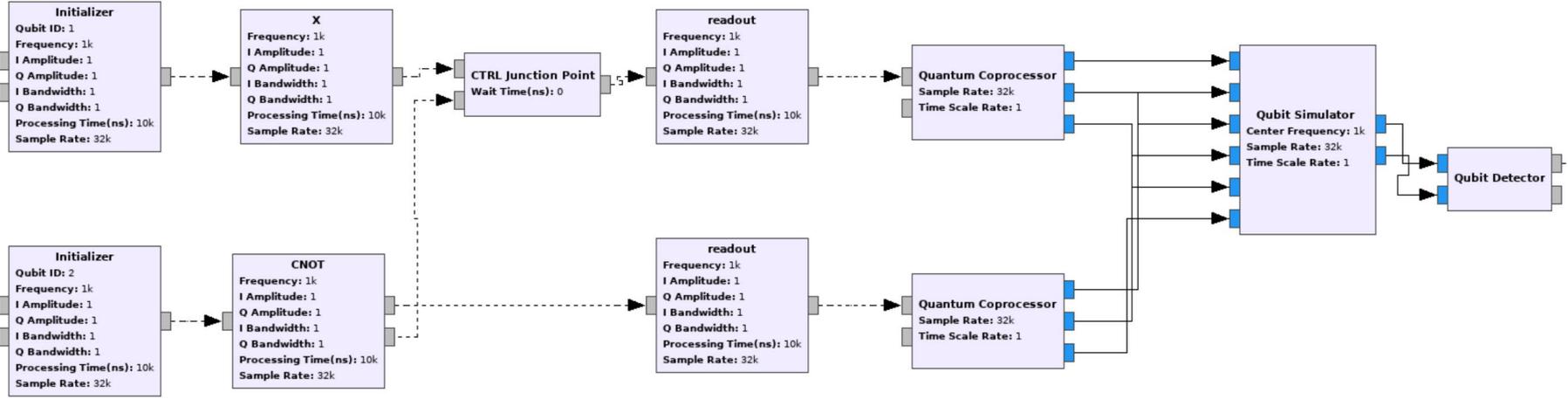
File Edit View Run Tools Help



1qubit_simulator_simple 1qubit_simulator_compiler 1qubit_simulator_SDR 2qubit_simulator_simple 2qubit_simulator_fullfunc

Options
ID: two_qubit_simulator_simple
Title: Simple 2Qubit Simulator
Author: Noritsu...siprop.org)
Generate Options: QT GUI

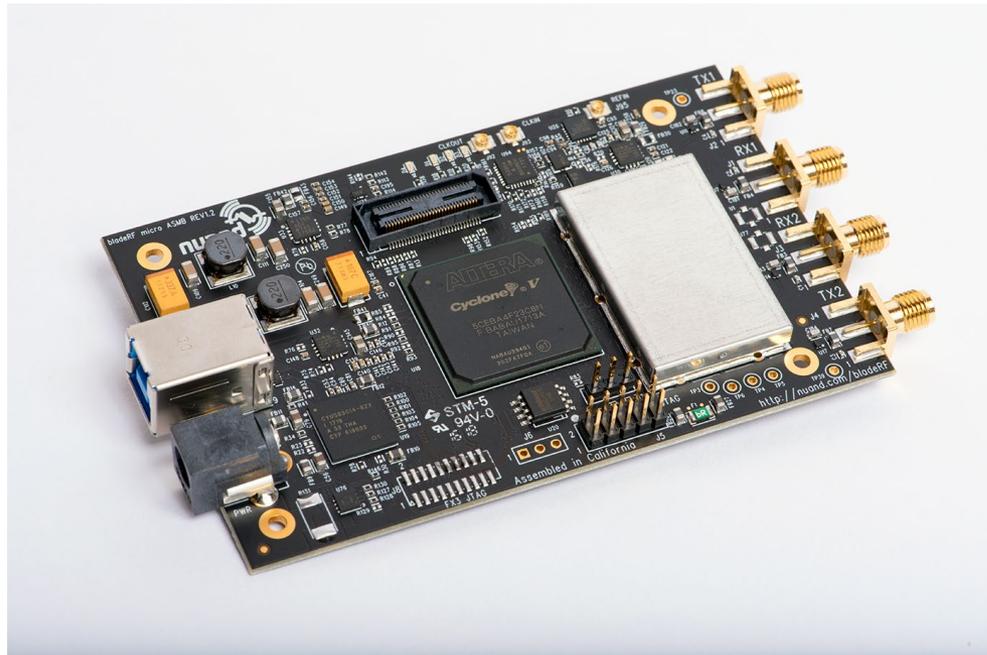
Variable
ID: samp_rate
Value: 32k



●デモ

おすすめ機器

- 必要機能
 - TX/RX が 2ポートずつ以上
 - 5GHz以上
- BladeRF2



- Demonstration of Two-Qubit Algorithms with a Superconducting Quantum Processor
 - <https://arxiv.org/abs/0903.2030>
- A Quantum Engineer's Guide to Superconducting Qubits
 - <https://arxiv.org/abs/1904.06560>
- An Experimental Microarchitecture for a Superconducting Quantum Processor
 - <https://arxiv.org/abs/1708.07677>
- Engineering the quantum-classical interface of solid-state qubits
 - https://tel.archives-ouvertes.fr/tel-00857654/file/thesis_Dewes_web.pdf
- Demonstrating Quantum Speed-Up with a Two-Transmon Quantum Processor
 - <https://tel.archives-ouvertes.fr/tel-00857654>
- 量子計算実験基礎
 - https://www.appi.keio.ac.jp/Itoh_group/abe/pdf/nquic.pdf