

IQ Demodulation for the Measurement of Quantum States

Kadir Akin

Senior Scientist
Quantum Engineering Center

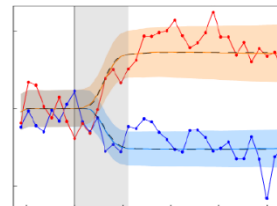
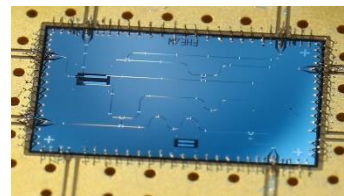
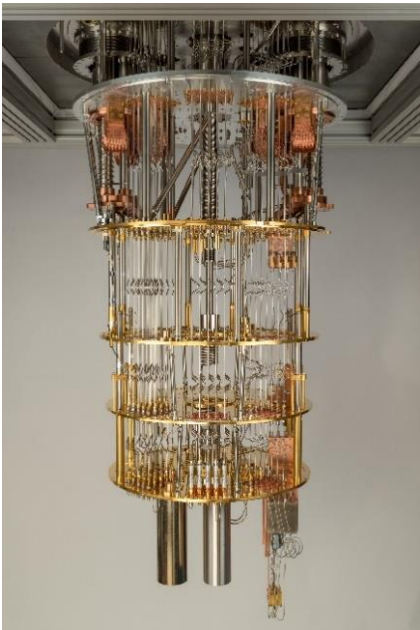
Office: HPF G7 ETH Hönggerberg

E-mail: aakin@phys.ethz.ch

Teaching Assistant: Michael Theodorou, Physics MSc

E-mail: mtheodorou@student.ethz.ch

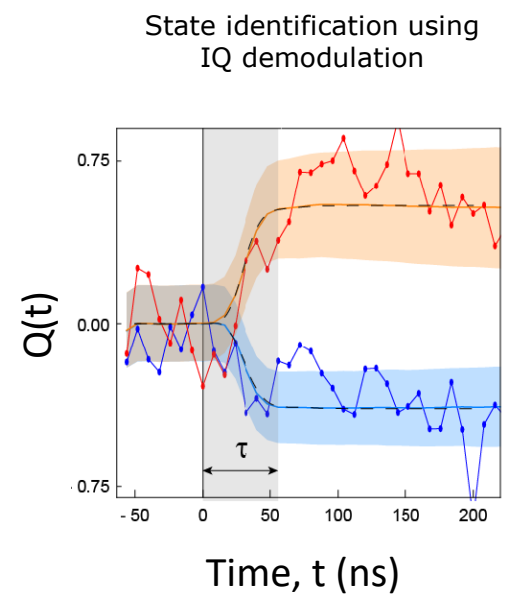
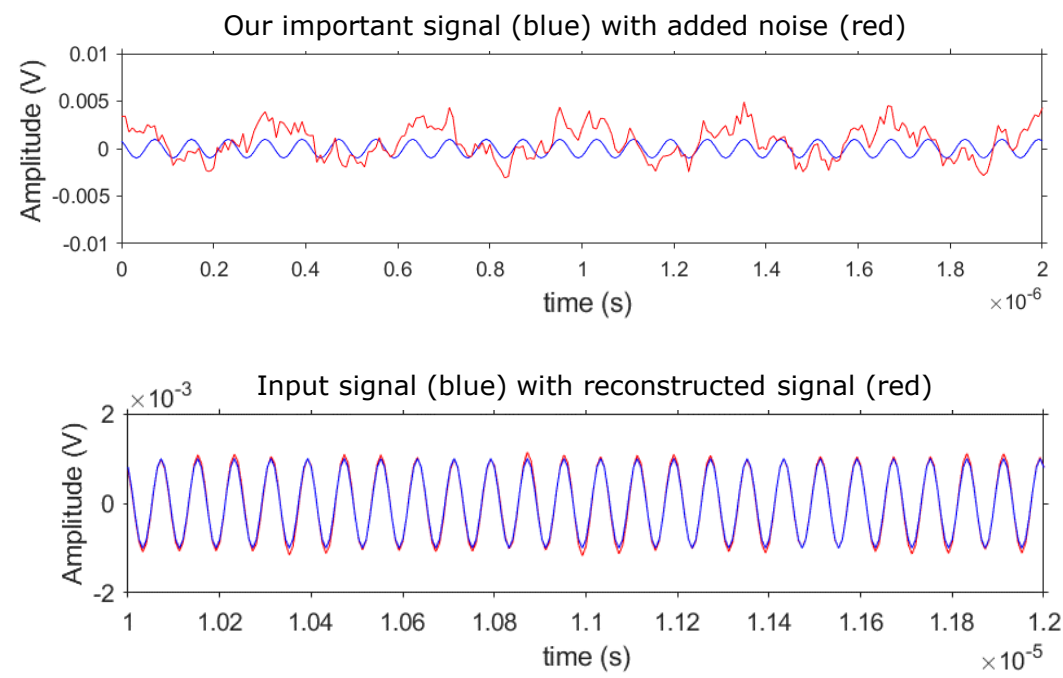
1.11.2023



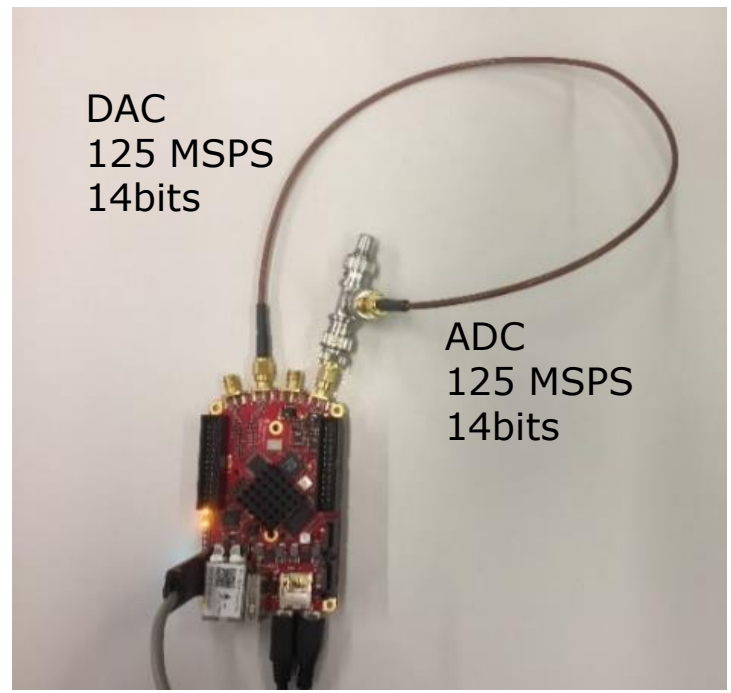
Today's Lecture: IQ Demodulation using FPGA and MATLAB

- Signals measured from quantum setup are often very noisy and mixed with different frequencies
- IQ demodulation is used to calculate the phase and amplitude of a signal to identify the quantum state

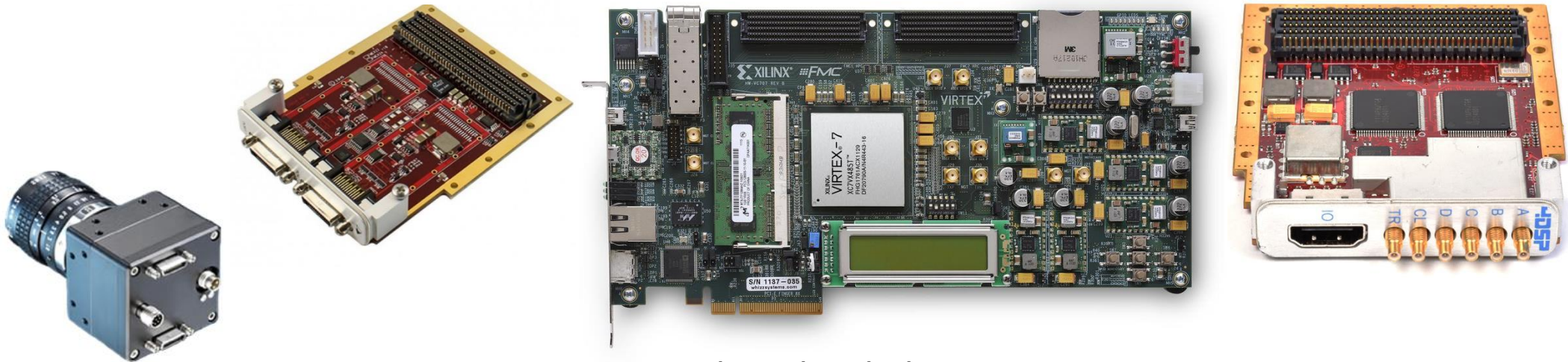
Lab Exercise with MATLAB



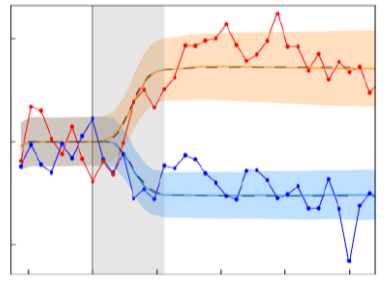
RedPitaya FPGA



Introduction to FPGA



VC707 board including Virtex-7 FPGA



- ❑ Massive parallel processing
- ❑ High memory bandwidth
- ❑ Faster than CPUs and GPUs
- ❑ Challenging to program and debug
- ❑ Applications: Video processing, machine learning, cryptography, database acceleration, quantum signal processing...

Why is FPGA Faster?

Example Code

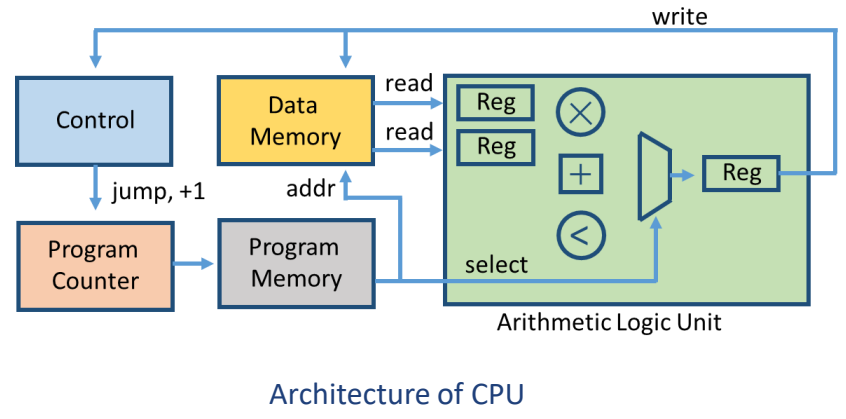
```

accum = 0;
for iteration = 1:100
{ c = coef(iteration);
  d = data(iteration);

  if (c < d)
    e = (c - 2) x (d + 3);
  else
    e = (d - 2) x (c + 3);

  accum = accum + e;
}
    
```

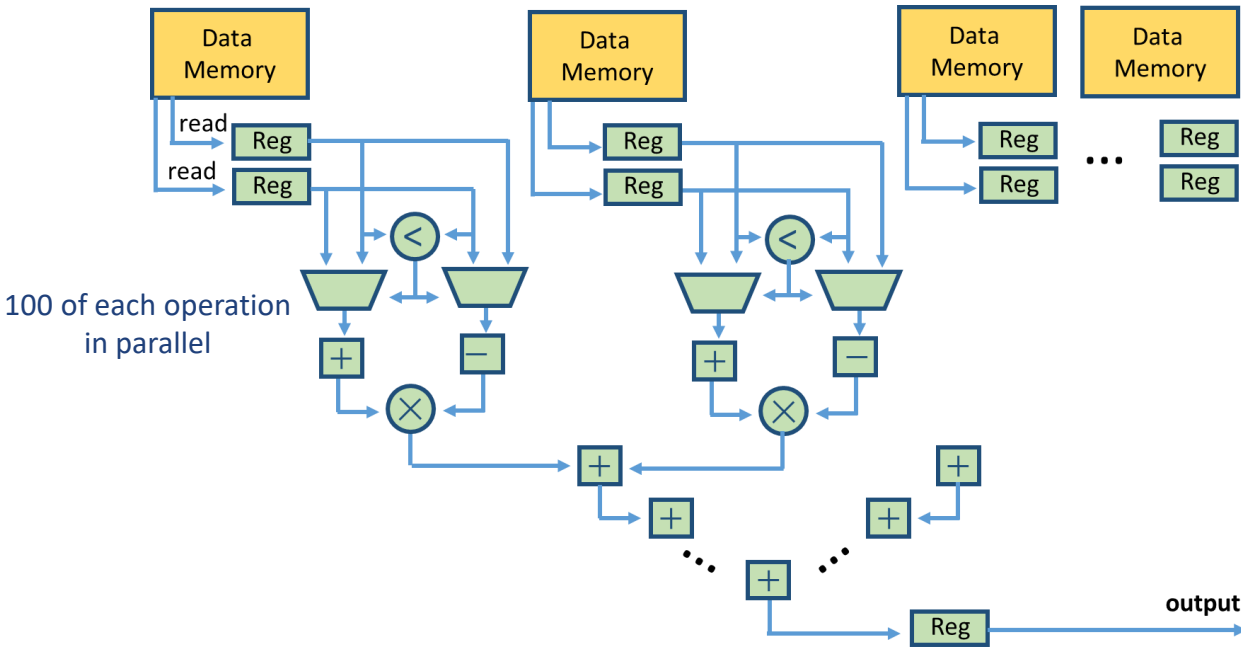
CPU



Architecture of CPU

vs

FPGA



- Execution loop over CPU (3000 clock cycles + Operating System)
- Limited Parallel Processing
- DDR4 offers 20 GB/s

⇔

- Configurable parallel architecture 3 clock cycles
- Massive Parallel Processing, for Virtex-7: 2800 Multiplier, 11200 32bits adder
- 1030 Block RAMs + DDR3 (5GB/s) offer 1726 GB/s

15min quantum measurement

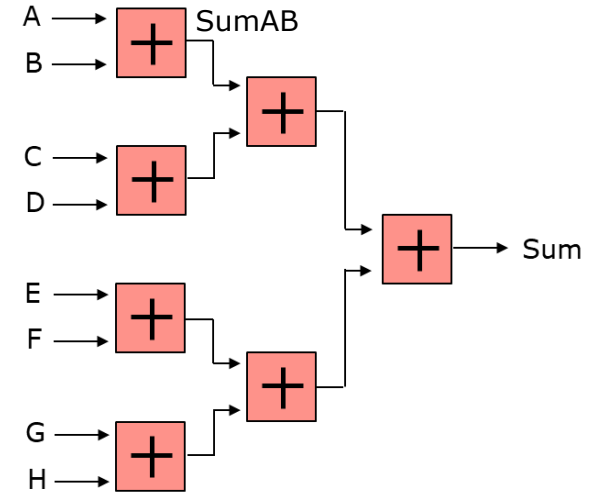
- 2.8TB memory in PC, 9 hours of processing
- Real-time, 260ns latency, 157 KB memory in PC

Your Verilog Code Describes your Hardware: Optimizations

```

assign SumAB = A + B;
assign SumCD = C + D;
assign SumEF = E + F;
assign SumGH = G + H;
assign SumABCD = SumAB + SumCD;
assign SumEFGH = SumEF + SumGH;
assign Sum = SumABCD + SumEFGH;
    
```

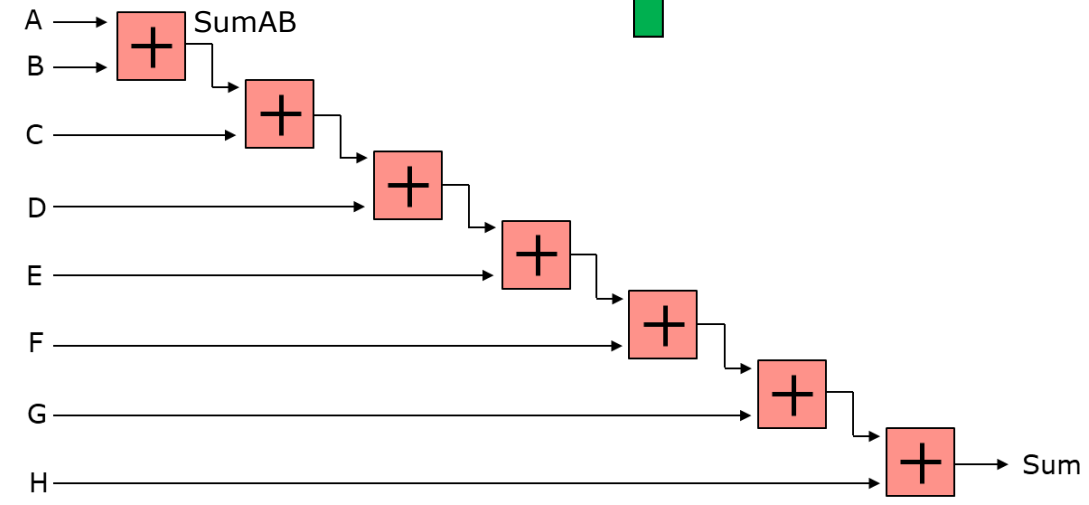
Verilog



```

assign SumAB      = A      + B;
assign SumABC     = SumAB  + C;
assign SumABCD    = SumABC + D;
assign SumABCDE   = SumABCD + E;
assign SumABCDEF  = SumABCDE + F;
assign SumABCDEFG = SumABCDEF + G;
assign Sum        = SumABCDEFG + H;
    
```

Verilog



More Efficient
in terms of latency

Xilinx Tools for Hardware-Software Co-design

Write Verilog Files
Vivado

Simulate Design
Vivado

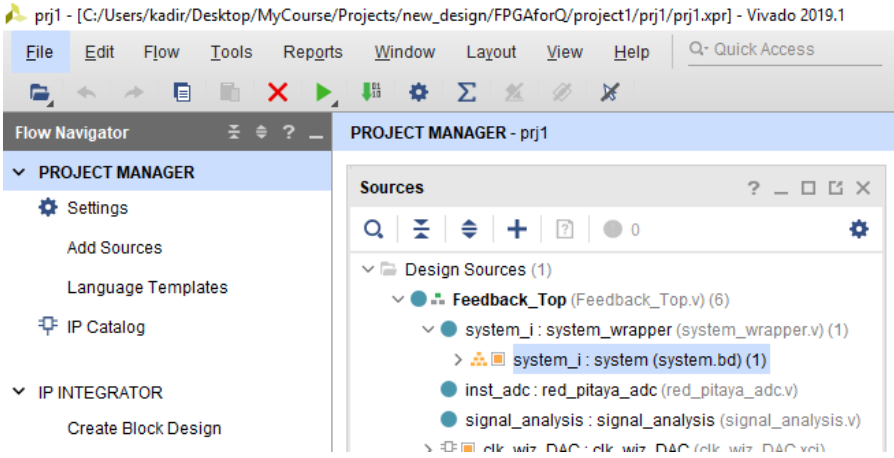
Synthesize Design
Vivado

Code C
SDK

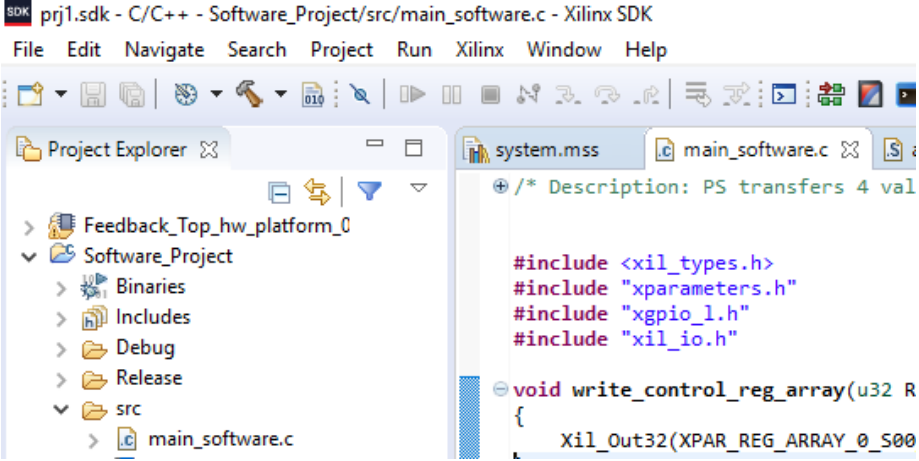
Program FPGA
SDK

Software Debugging
SDK

Hardware Debugging
Vivado



Xilinx Vivado

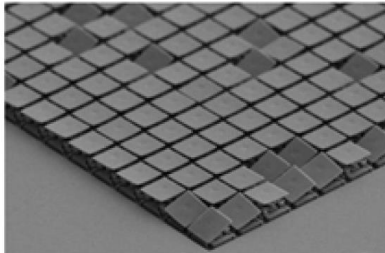
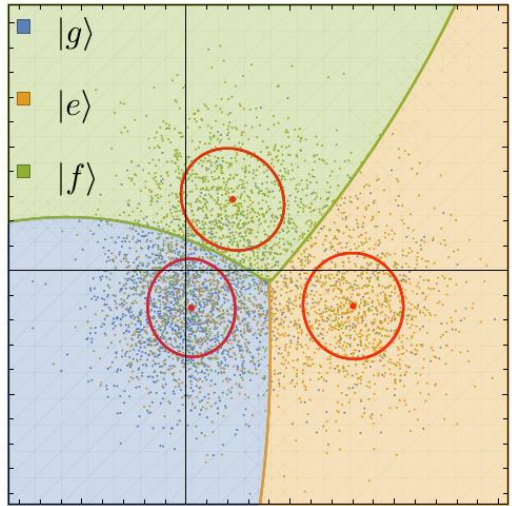
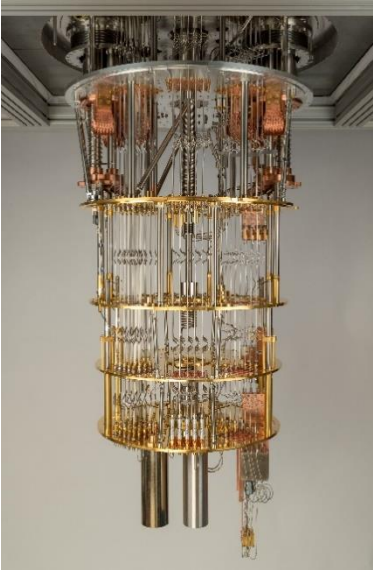


Xilinx SDK

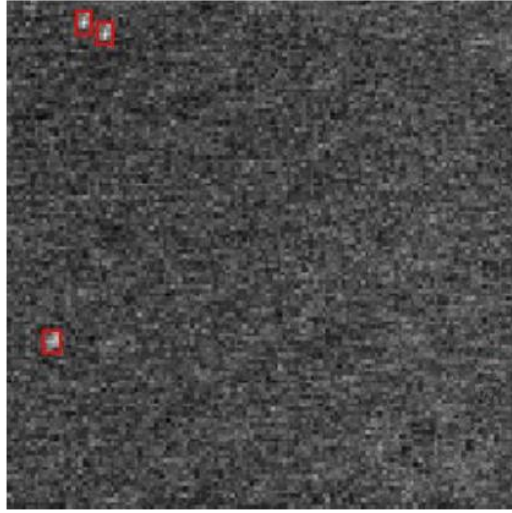


FPGA in Quantum Experiments

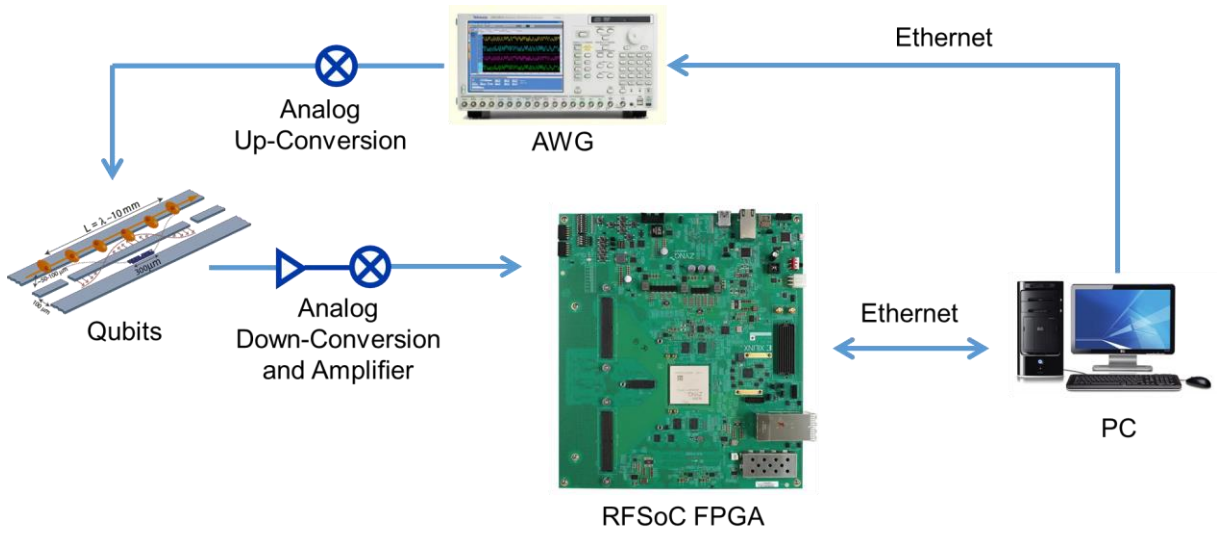
- DSP for Superconducting Qubits
 - Overview of experiment setup
 - User applications (averager, histogrammer, correlator, state identification, machine learning)
- Image Processing for Ion-Traps



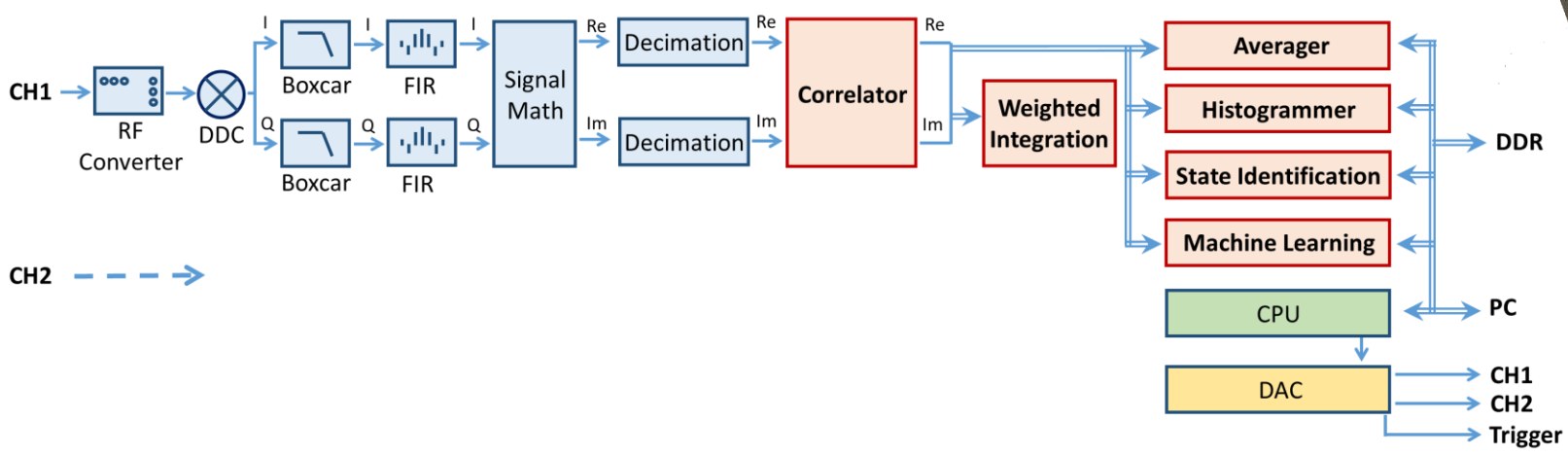
Digital Micromirror Device



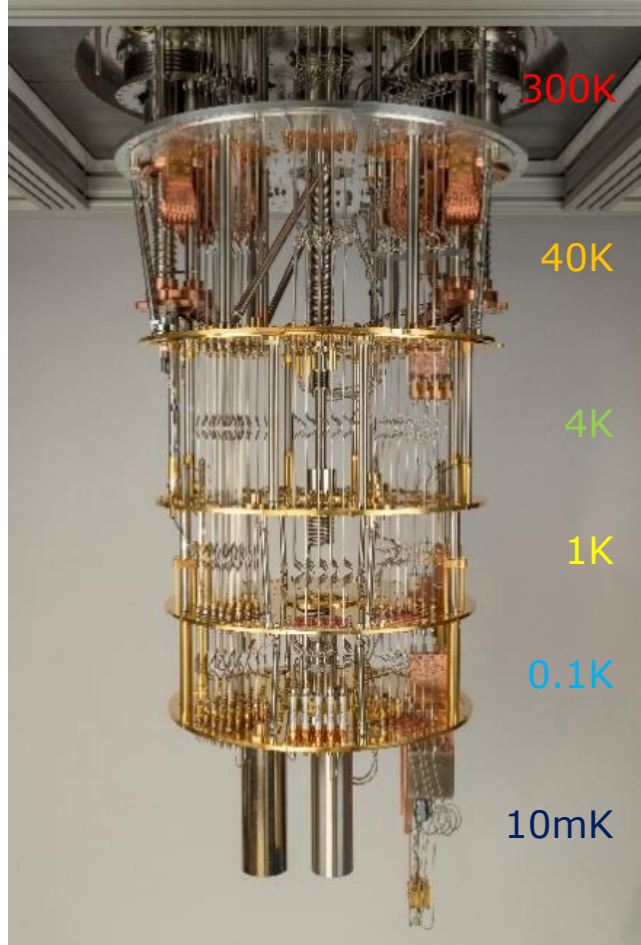
Overview of Quantum Experiment Setup



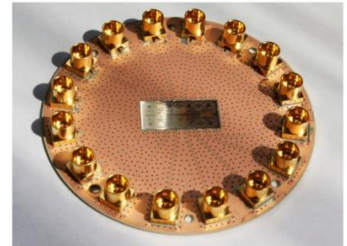
- 16 ADC (@ 2.5 GSPS) and 16 DAC (@ 10 GSPS) Channels
- Real-time and low-latency
- ❑ Feedback (1 us latency is required):
 - Impossible with PC → 260ns latency with FPGA



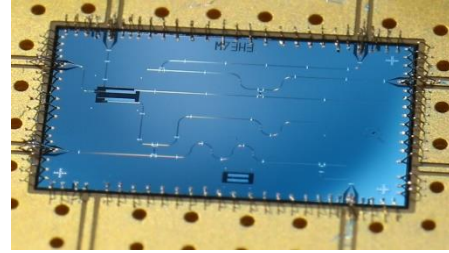
Overview of Quantum Experiment Setup



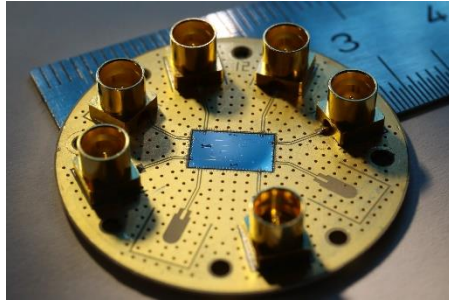
Cryostat



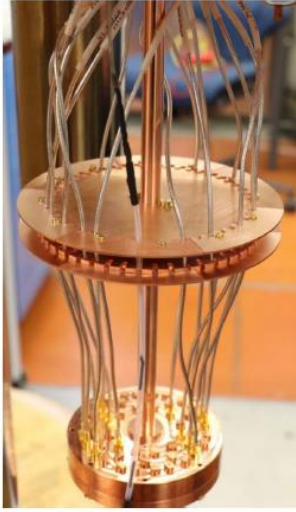
Qubit chip with connectors



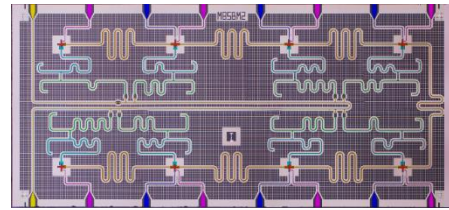
Qubit chip



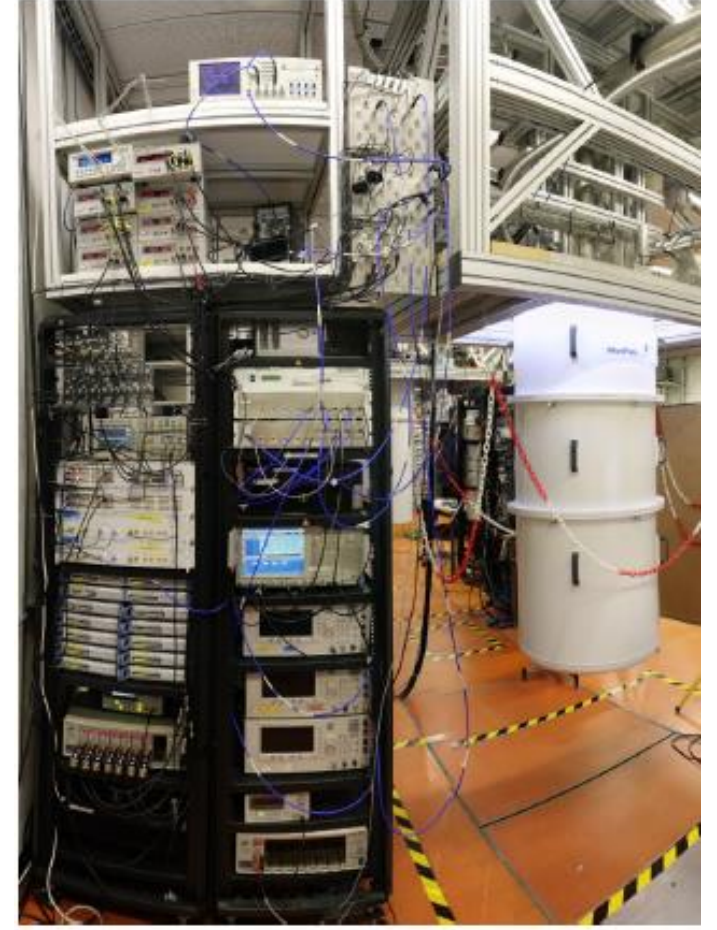
Qubit chip with connectors



PCB holder



Qubit chip design

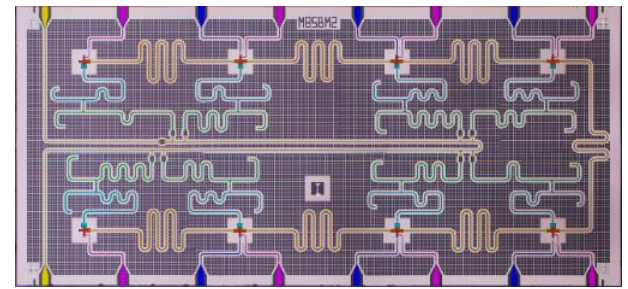
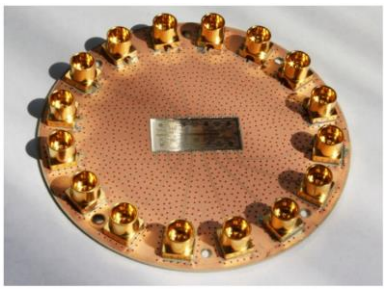
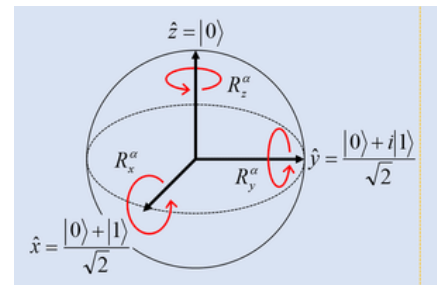
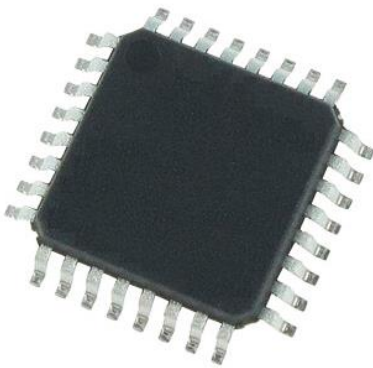
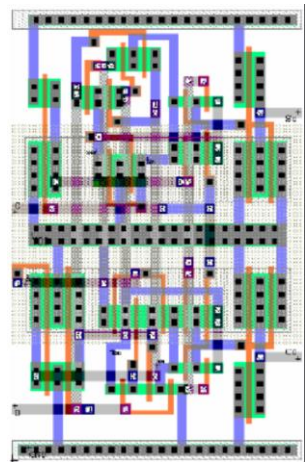
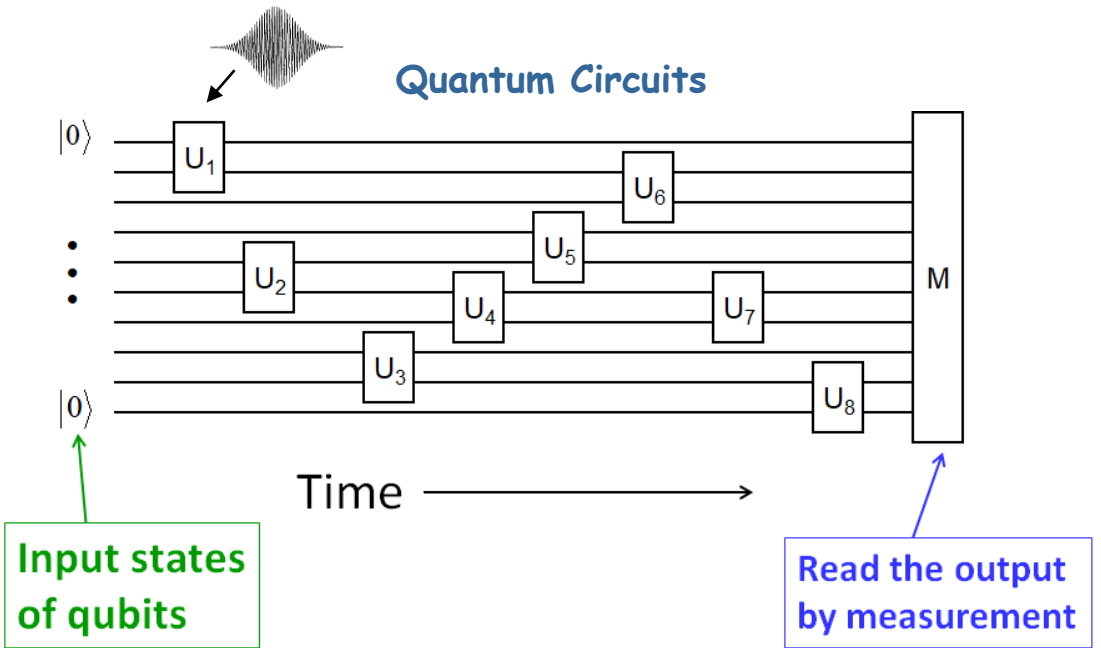
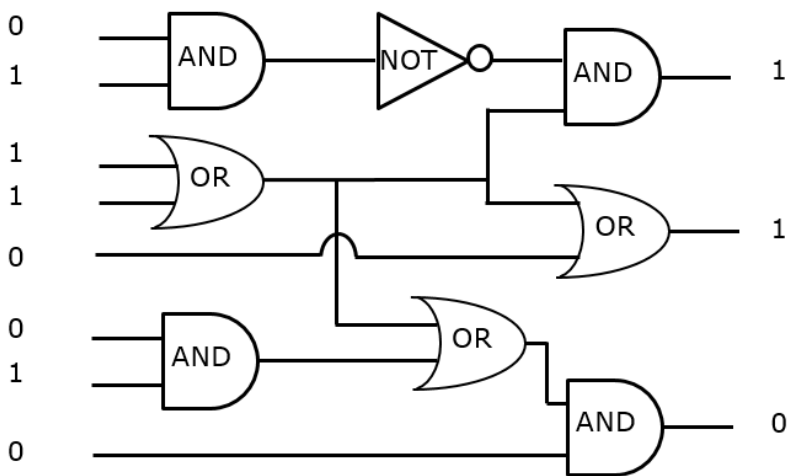


Qudev Lab

Team of Expertise: Quantum Theory, Cryostats and Cryogenics, Fabrication, RF Circuits, Analog Circuits, Digital Signal Processing, FPGA, PCB design, Software Engineering, Computer Network

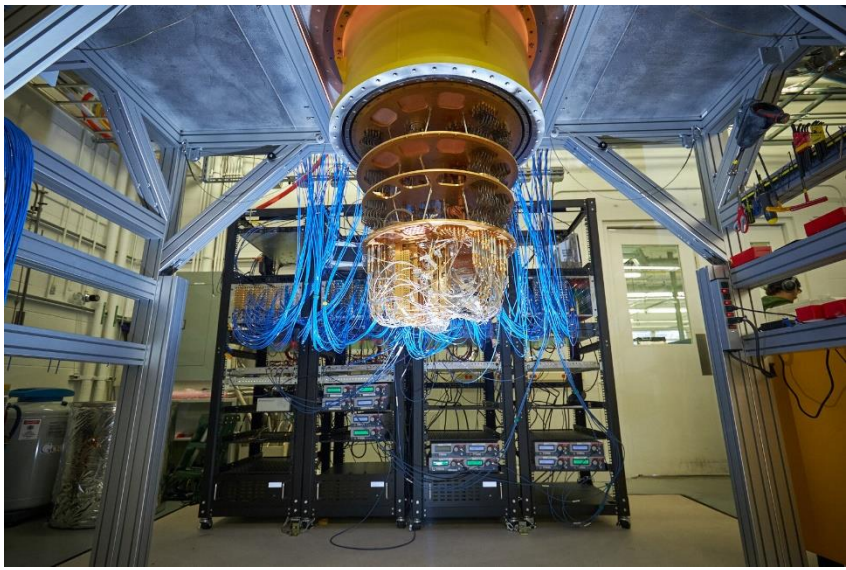
Comparison between Quantum and Digital Circuits in terms of Gate Operations

Operations in classical computer

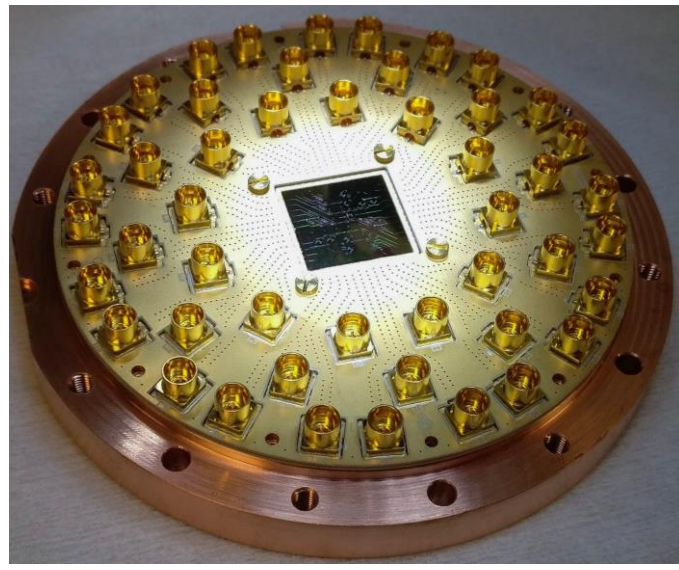


Requirement: Scalable System for Controlling and Measuring High Number of Qubits

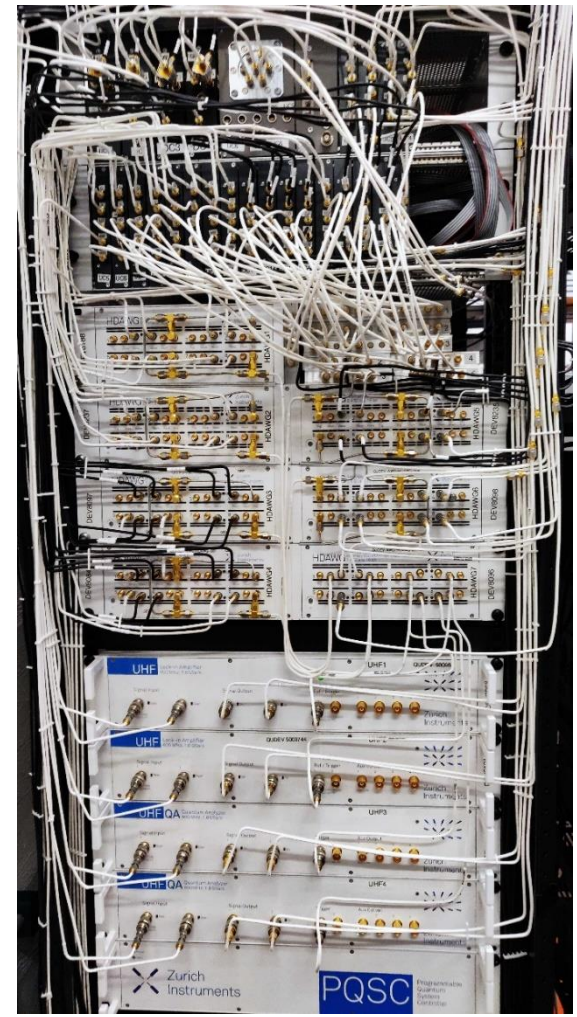
100 qubits would require around:
500 fast ADC and DAC channels (1-5 GSPS)
500 microwave generator channels (4-20 GSPS)



Google's Quantum Computer supporting 10 qubits



Control and measurement electronics used for 17 qubits of Qudev



Requirement: Scalable System for Controlling and Measuring High Number of Qubits

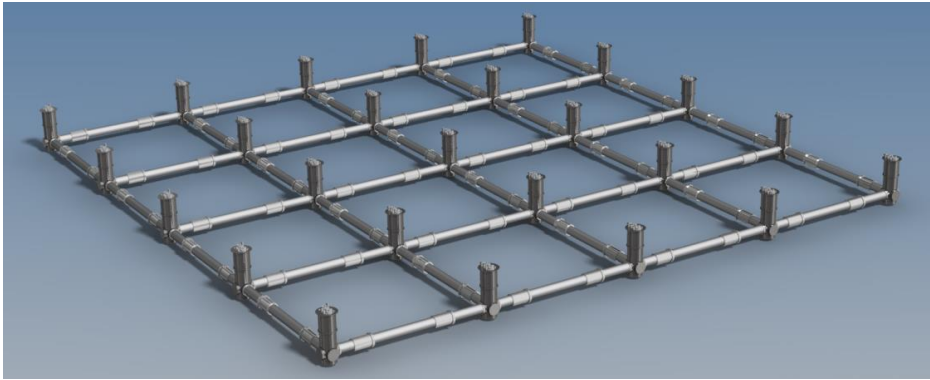
Synchronization of multiple systems for network of qubits



Ref: P. Magnard, et al., Physical Review Letters, 125(26), 260502, 2020

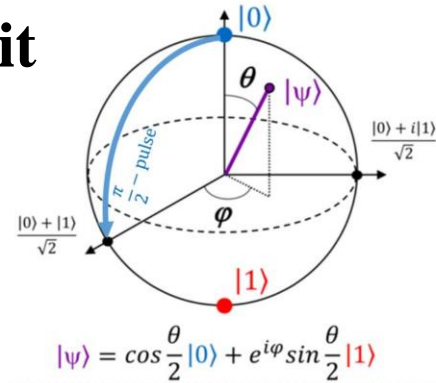
Cryo Link in Qudev

Network of Quantum Computers in Future

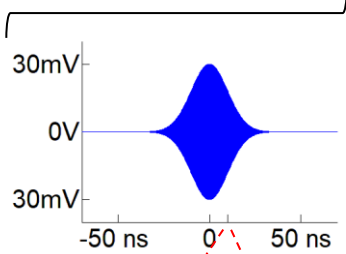


Ref: Simon Storz

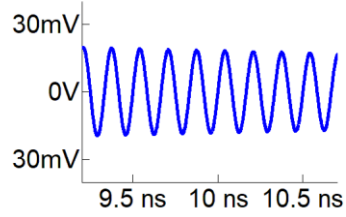
Control and Measurement of a Qubit



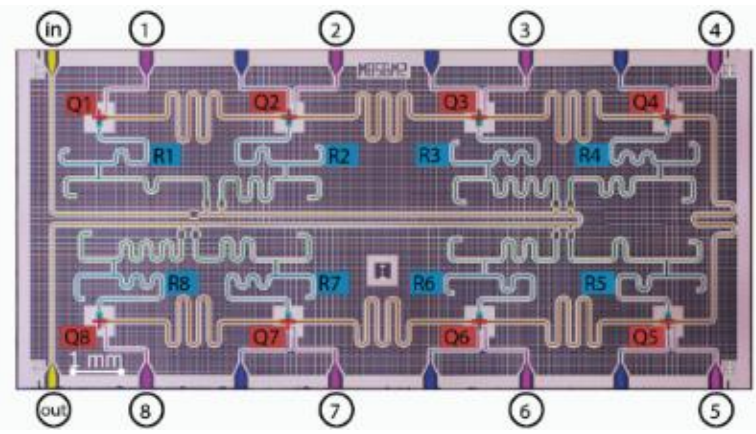
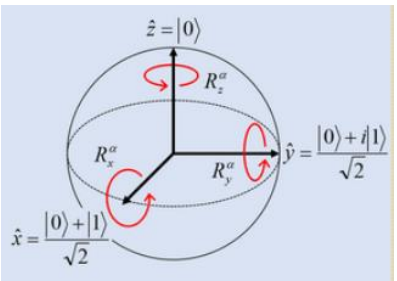
Control



5-10 GHz Sin Wave with Gaussian Envelope at Qubit Frequency

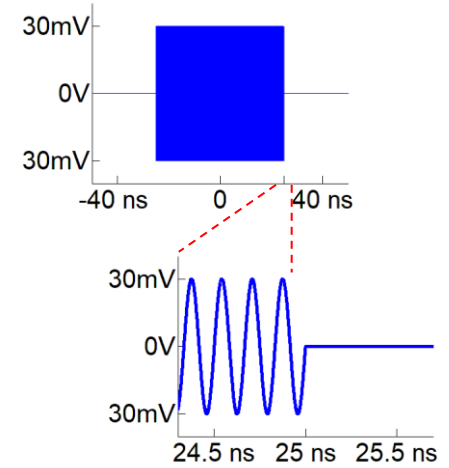


XY control

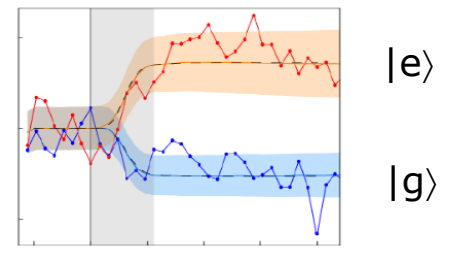
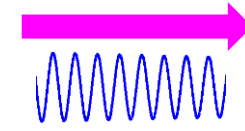


Measurement

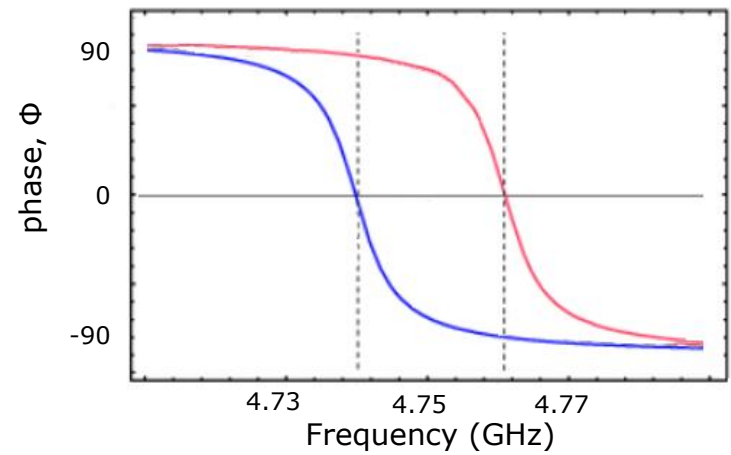
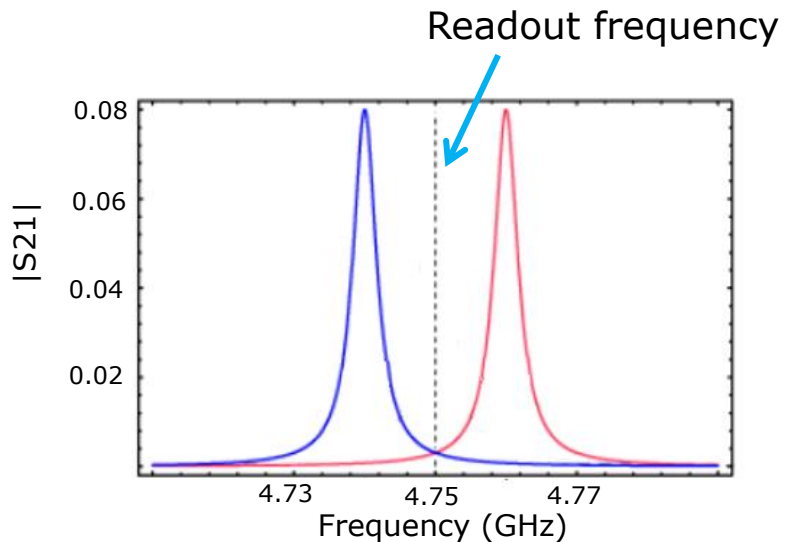
5-10 GHz Sin Wave with Rectangular Envelope at Resonator Frequency



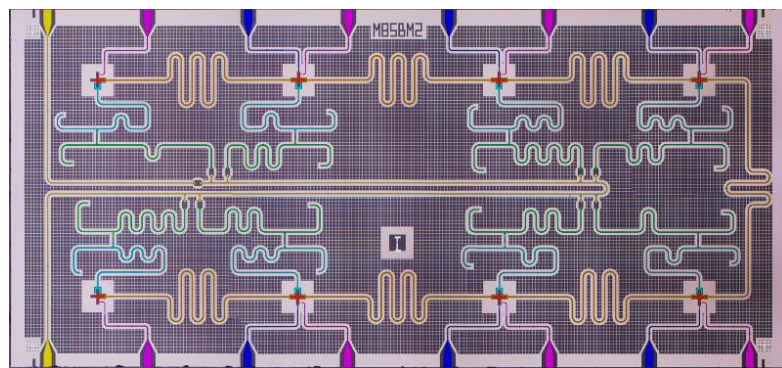
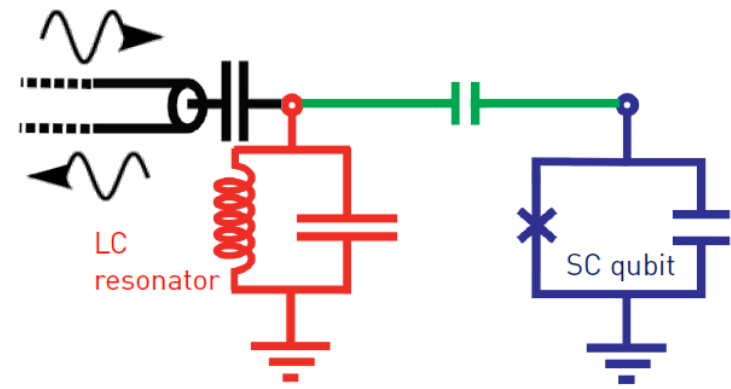
Quantum state presents in amplitude and phase response



Measurement of a Superconducting Qubit



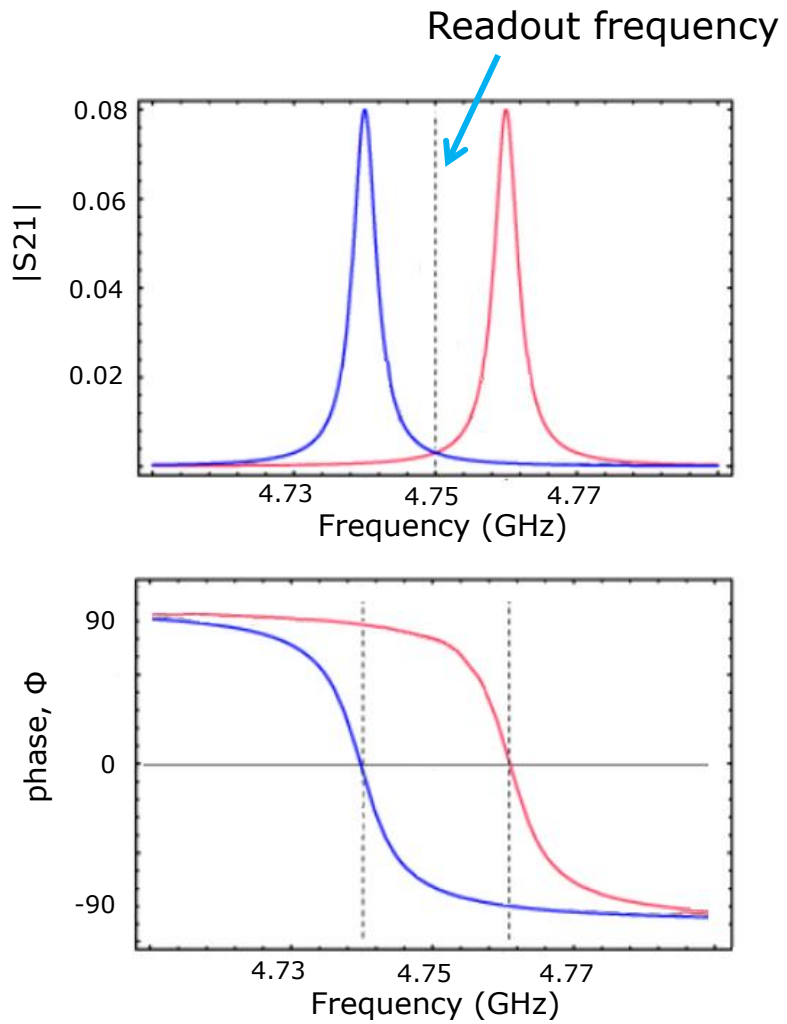
A readout of a single qubit circuit with an LC readout resonator (without purcell filter), amplitude and phase response



- Qubits
- Readout resonators
- Purcell filters
- Coupling Bus resonators
- Charge lines
- Flux lines
- Feed line

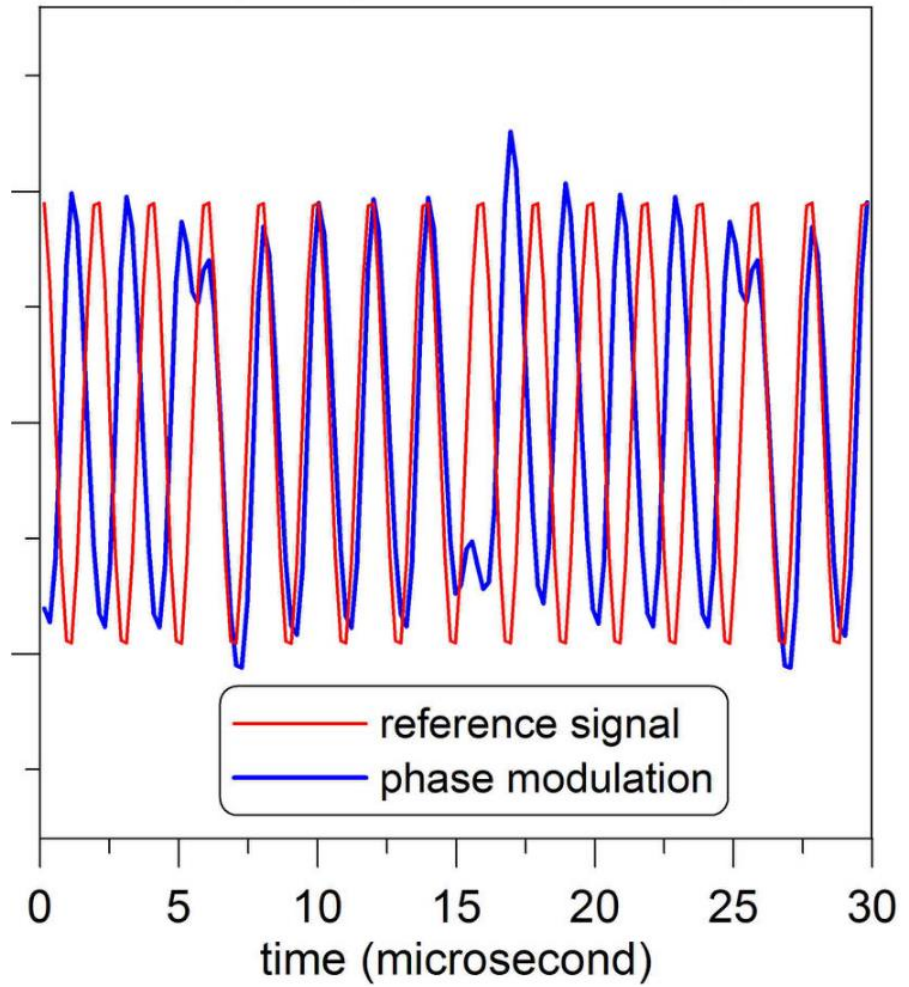
J. Heinsoo et al., Phys. Rev. Applied 10, 034040 (2018)

Quantum Mechanics with Superconducting Qubits



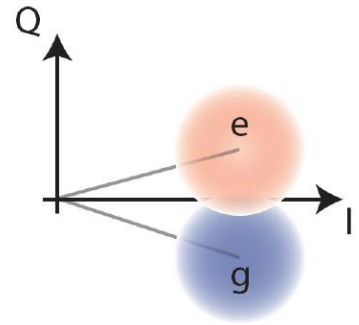
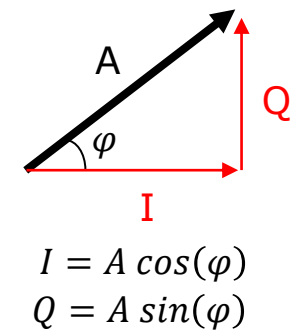
A readout of a single qubit circuit with an LC readout resonator (without purcell filter), amplitude and phase response

$$A(t) \cos(2\pi f t + \varphi(t))$$

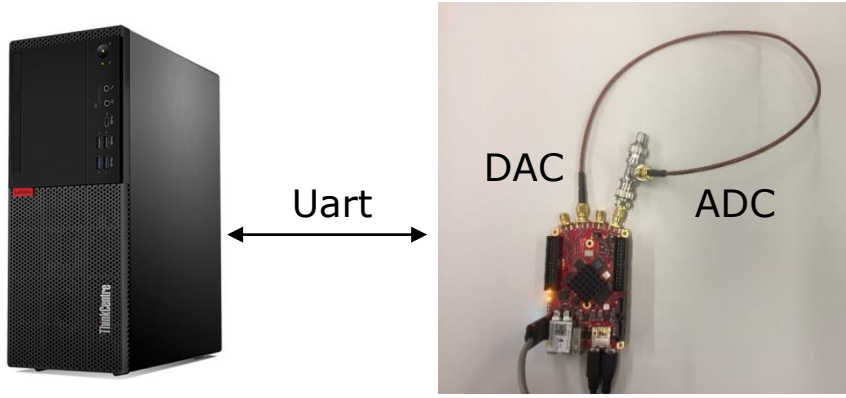
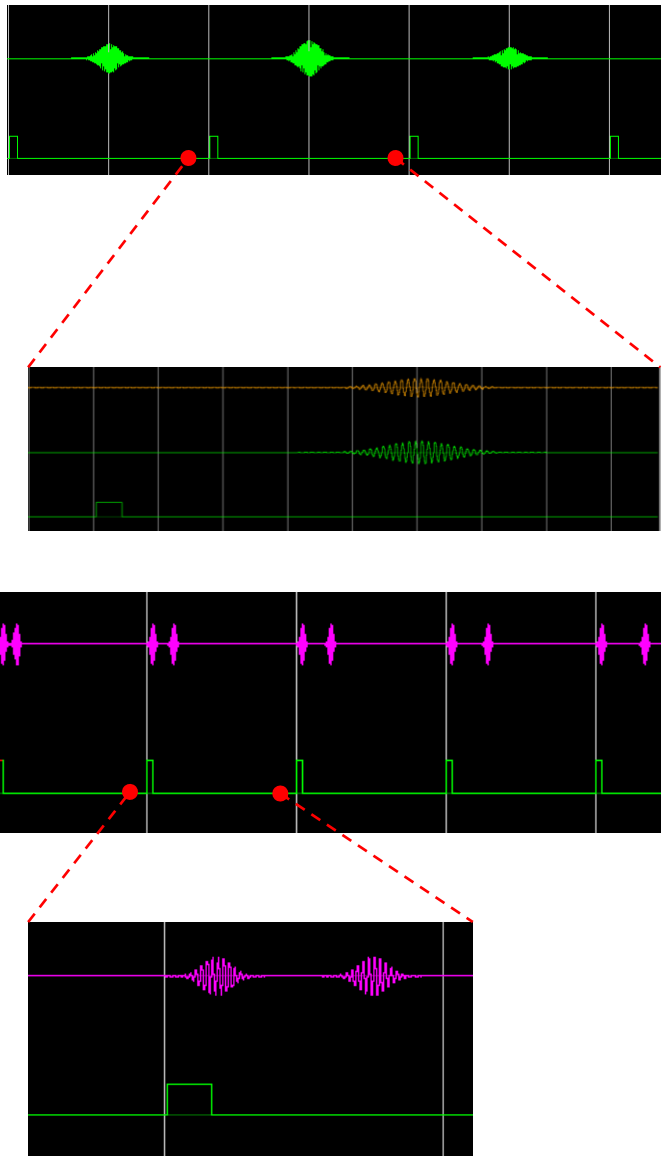
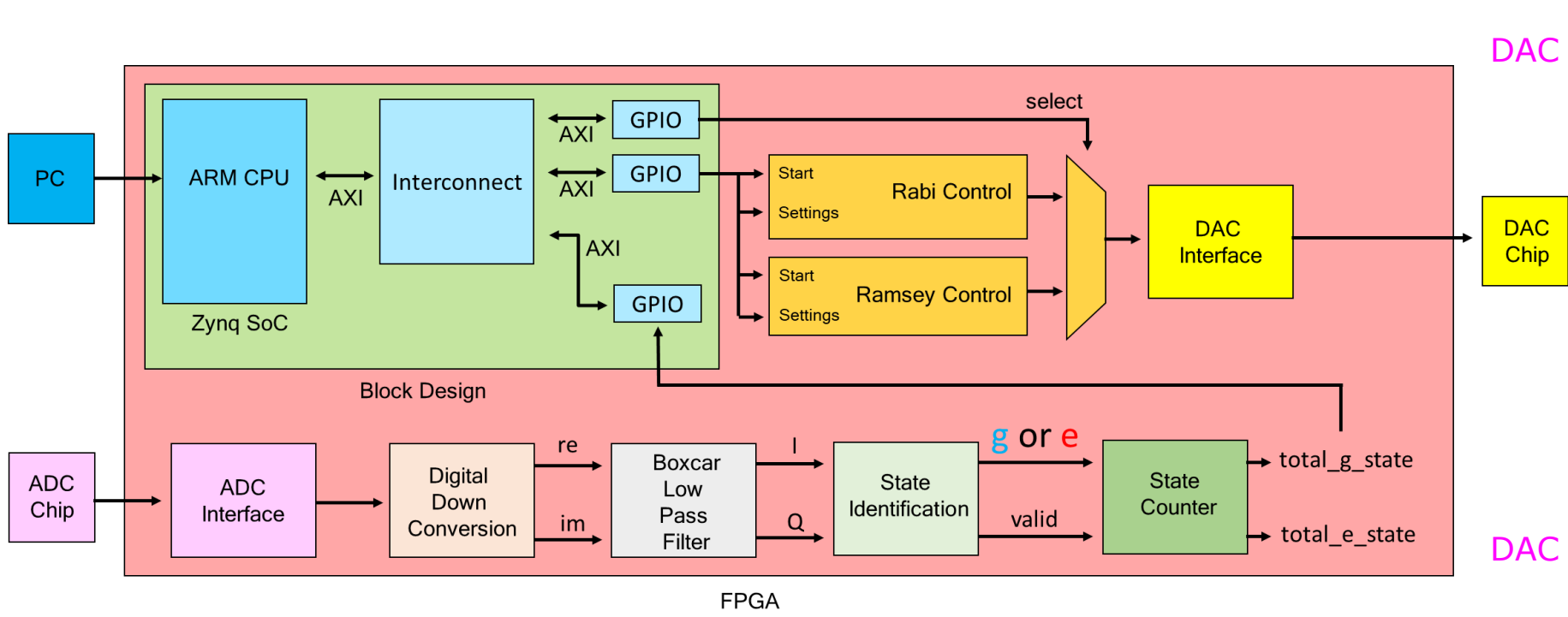


$$A e^{i\phi} = I + i Q$$

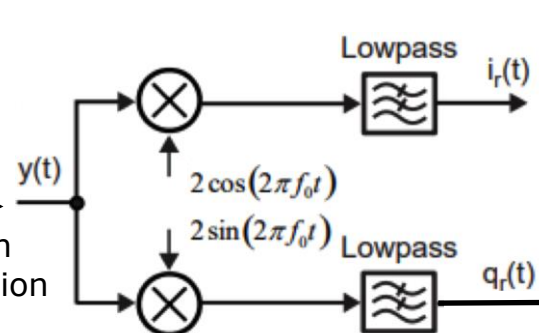
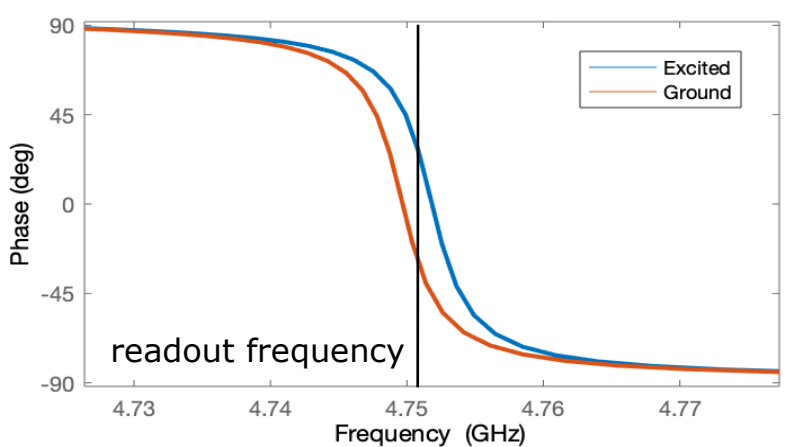
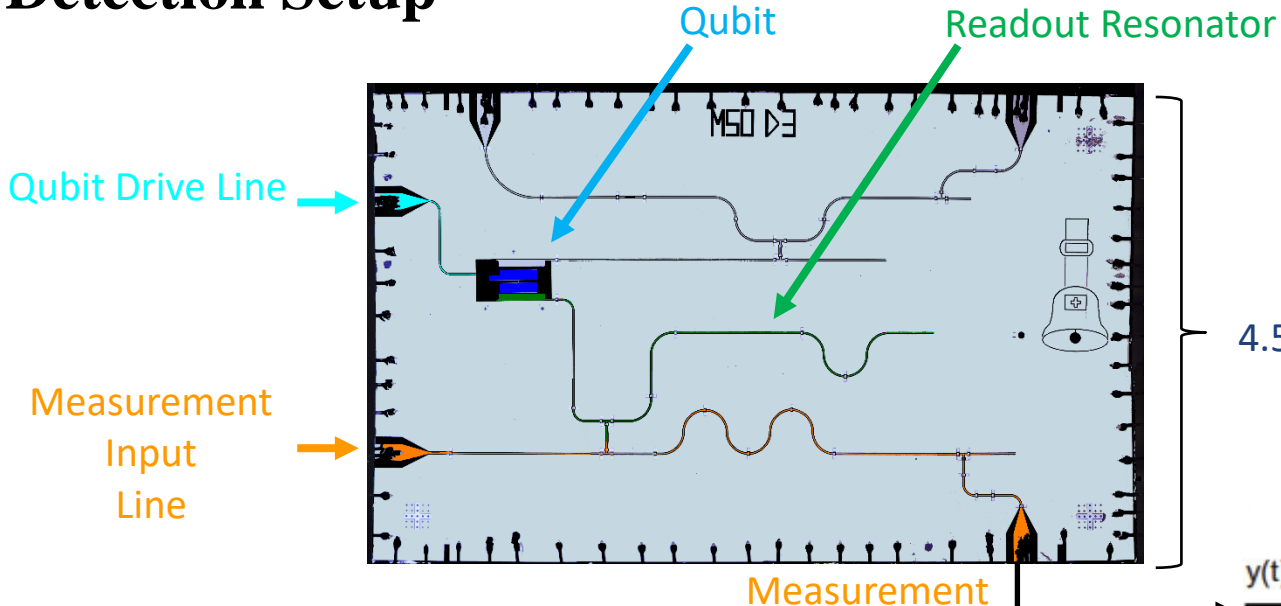
↑ signal amplitude ↑ Phase ↑ In-phase and quadrature components



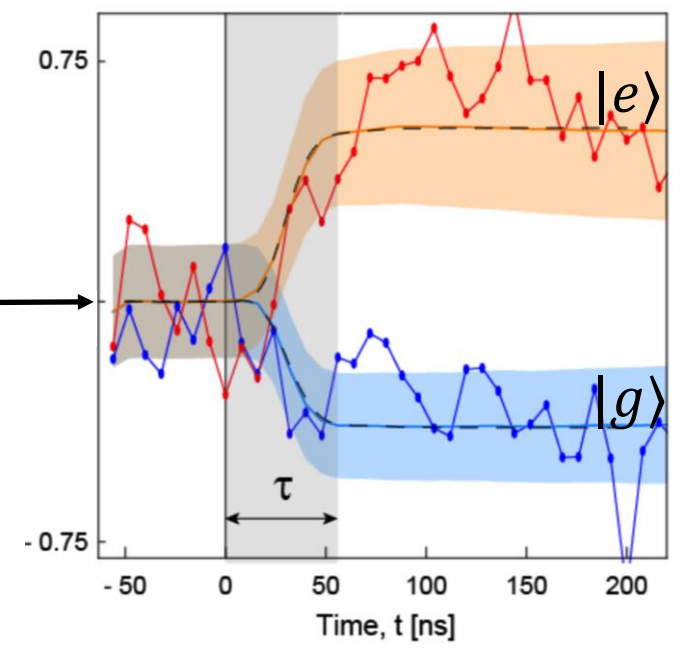
Digital Signal Processing for Quantum Experiments using FPGA



Detection Setup

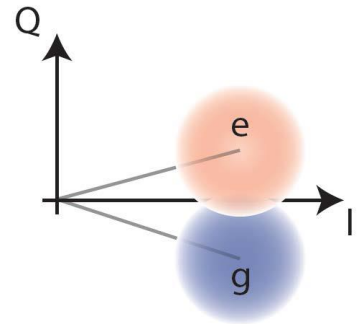
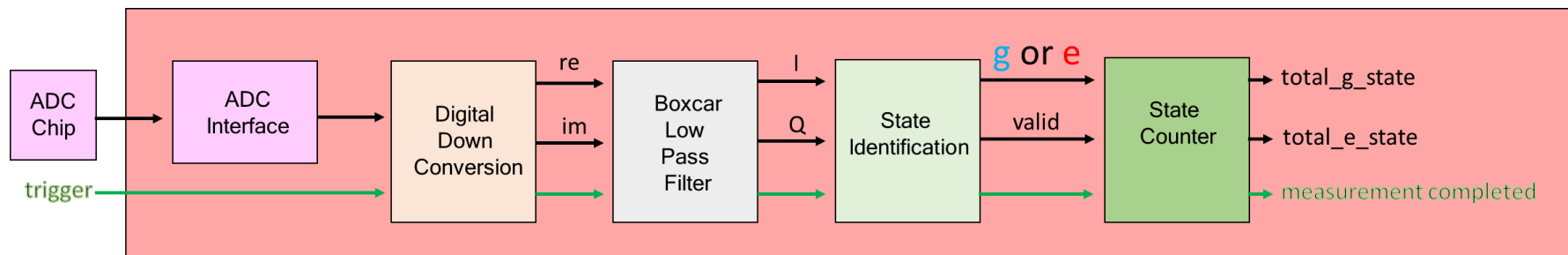


Time Integration
(adding-up consecutive values)

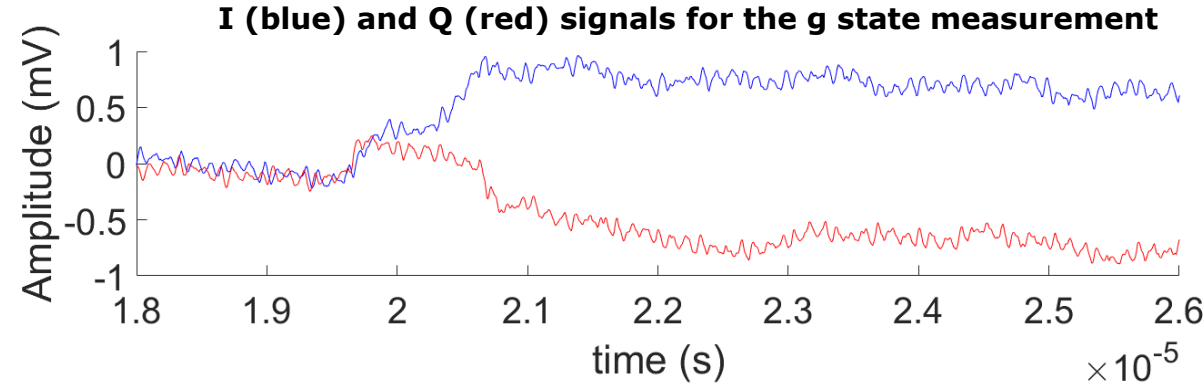
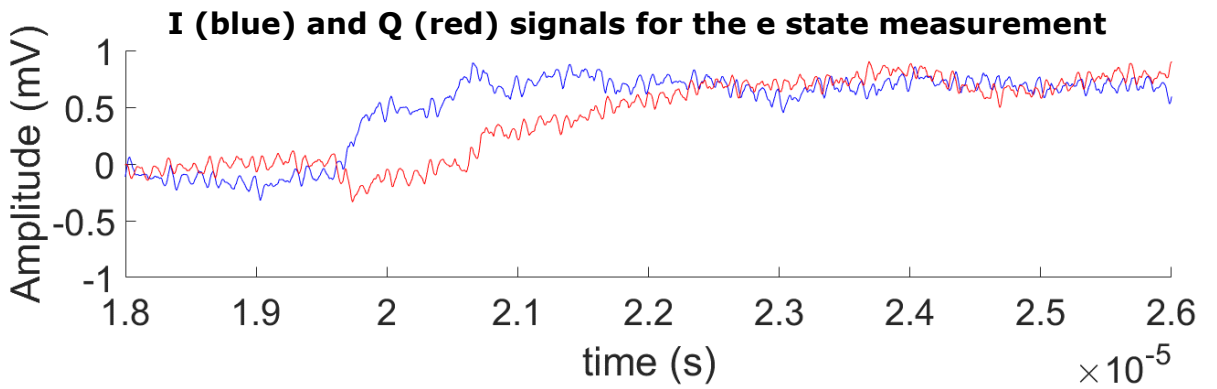
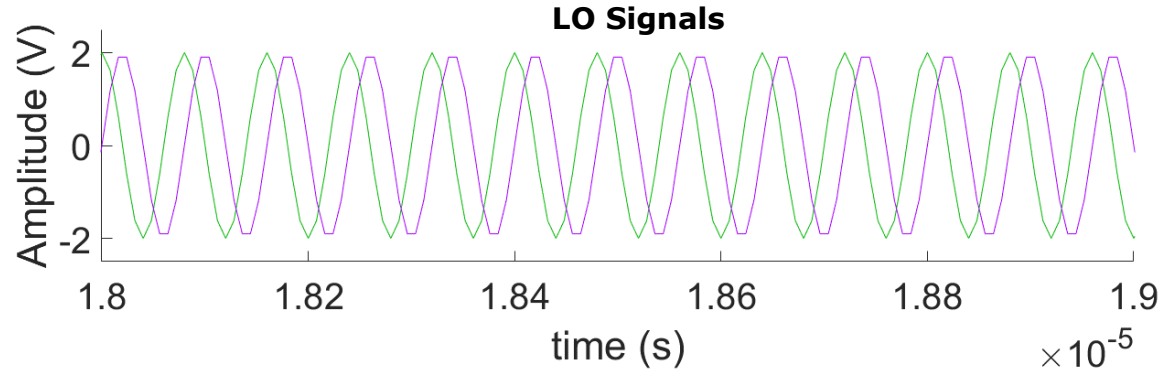
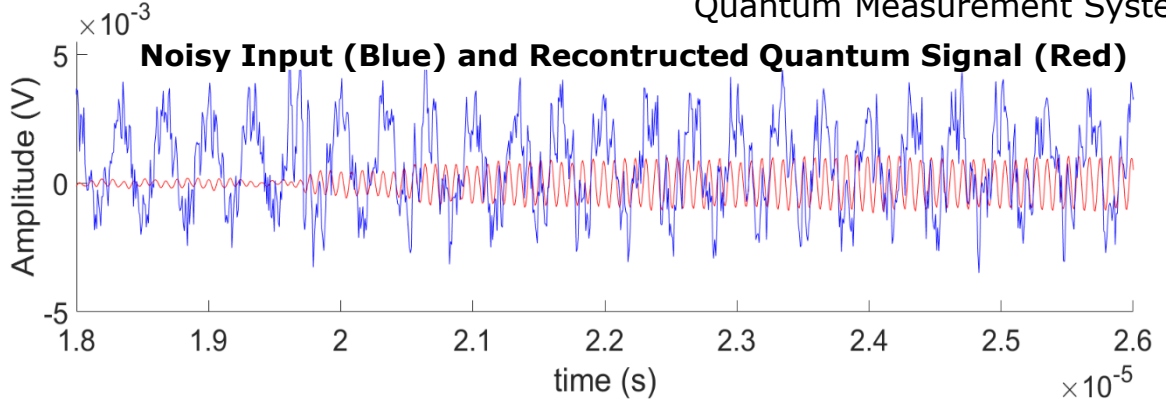


FPGA

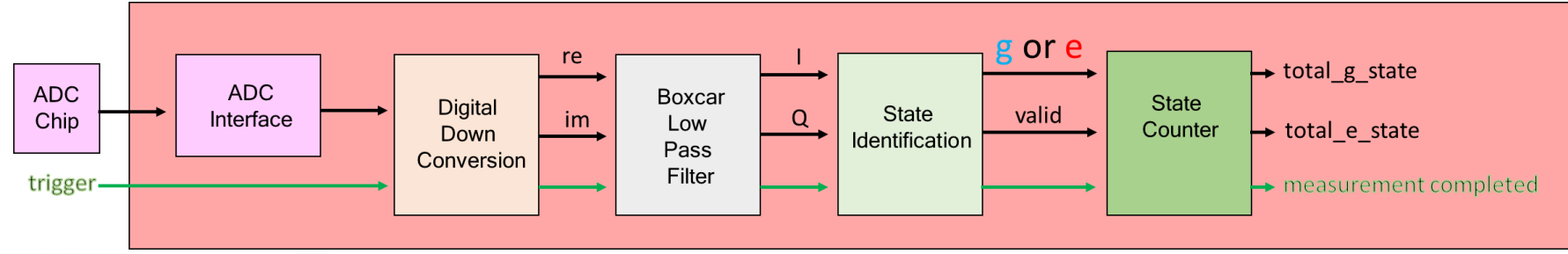
Digital Signal Processing for Quantum Experiments using FPGA: Measurement



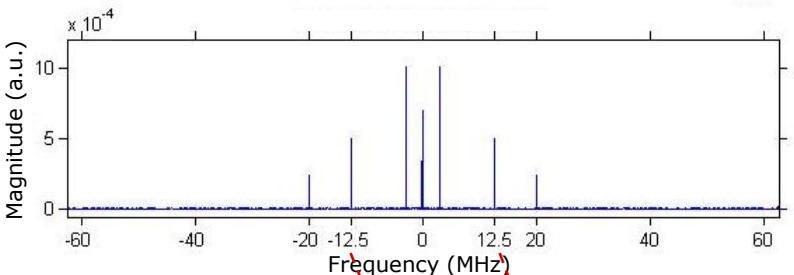
Quantum Measurement System in FPGA



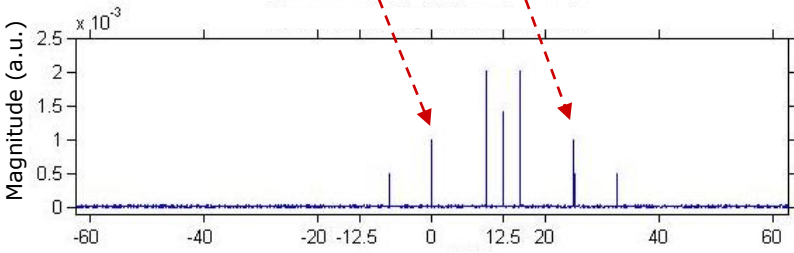
Digital Signal Processing for Quantum Experiments using FPGA: Measurement



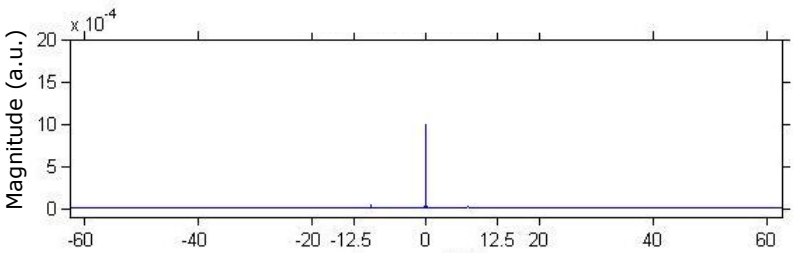
Quantum Measurement System in FPGA



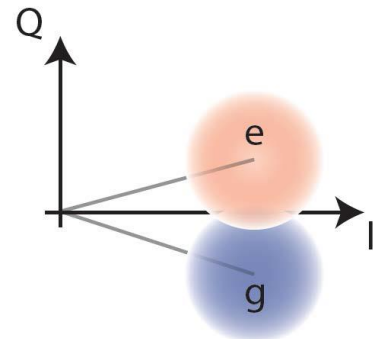
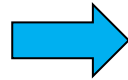
FFT of Input



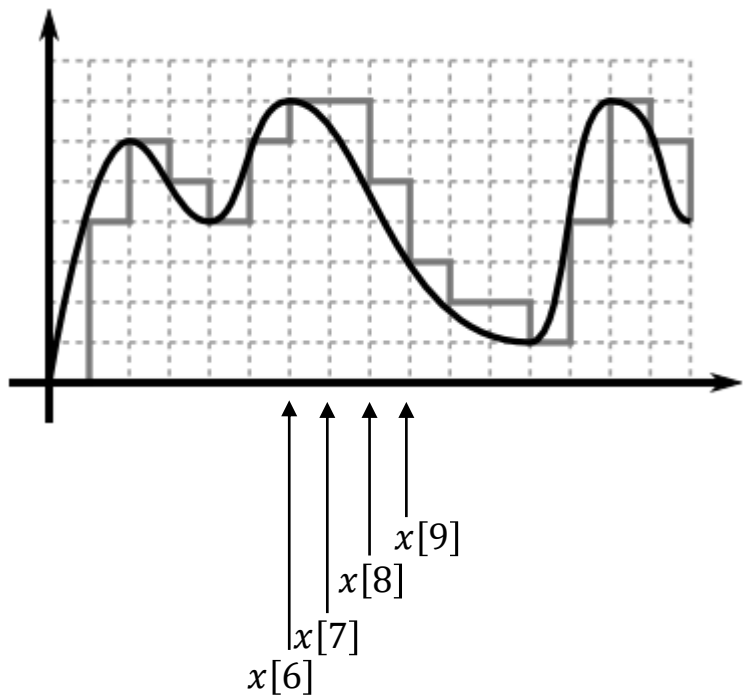
FFT after DDC



Removing noise and unexpected signals
FFT after Boxcar

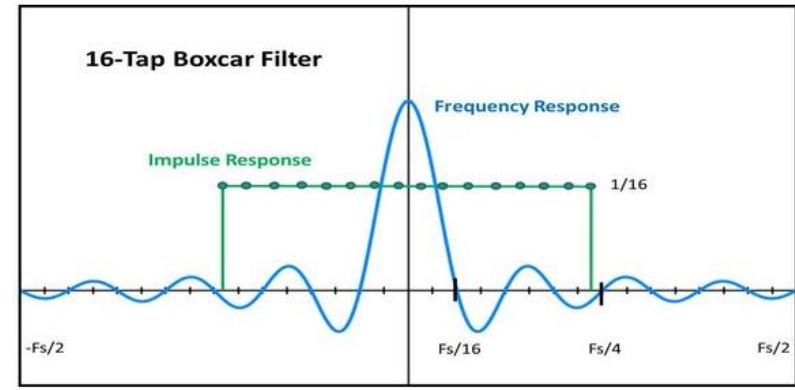
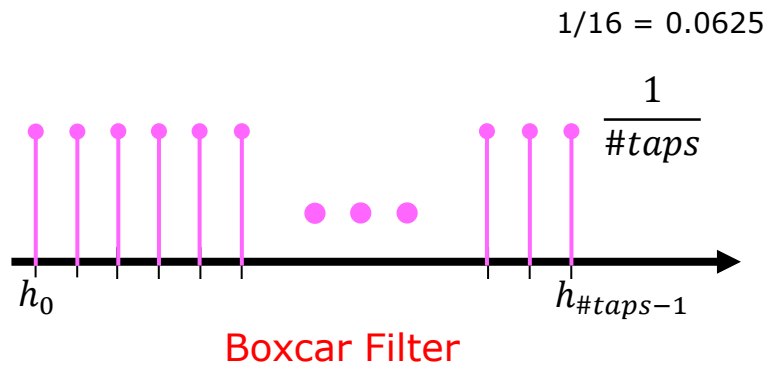
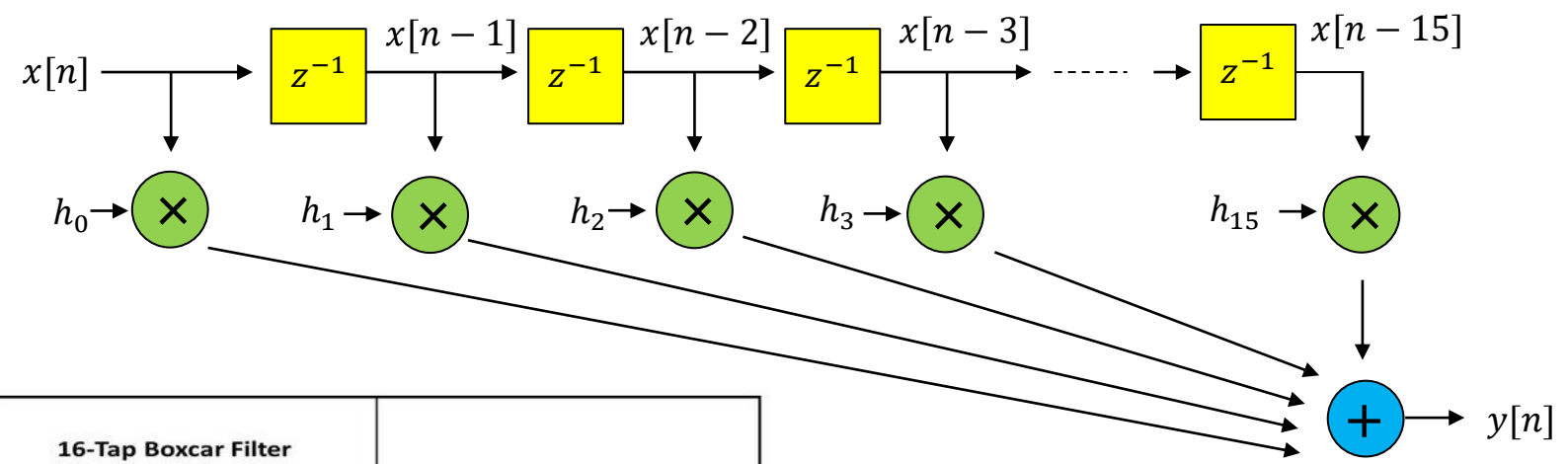


Discrete Time FIR Filter



x are inputs
 h are the weights
 y are the outputs
 z^{-1} are the delays (register)

$$y[n] = h_0 x[n] + h_1 x[n - 1] + \dots + h_N x[n - N]$$



$F_s = 125 \text{ MHz}$ (samples every 8ns)
 $F_s/16 = 7.8 \text{ MHz}$

Calculating Correlations with Histogrammer

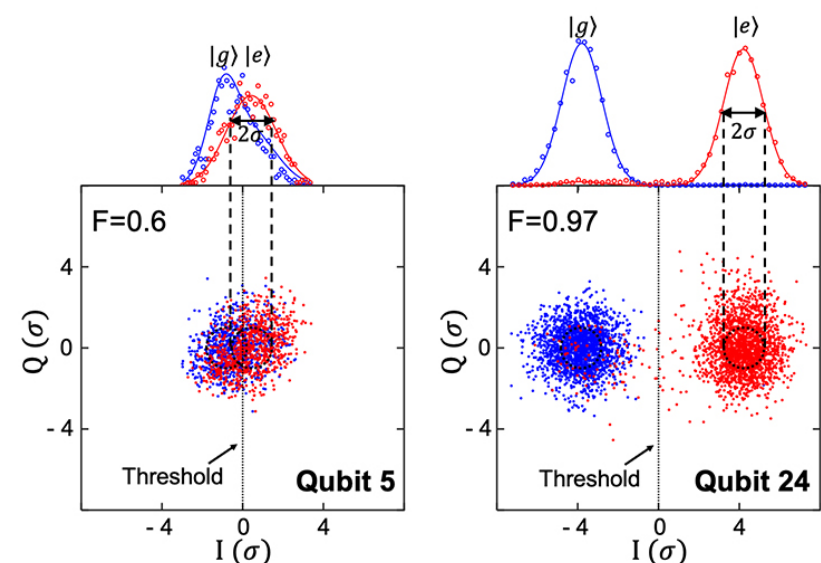
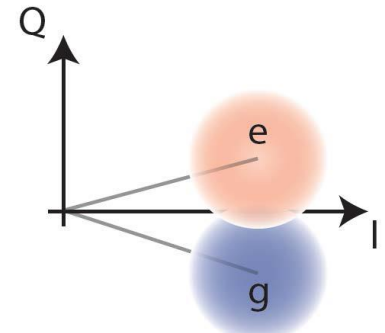
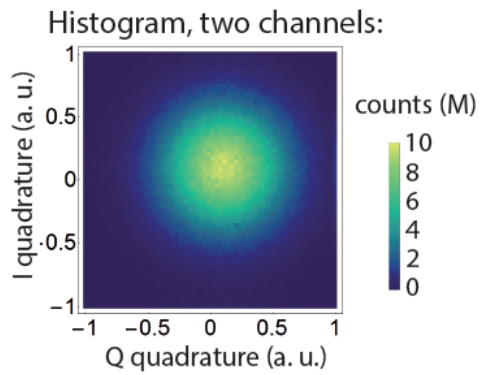
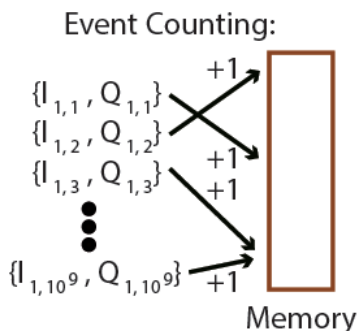
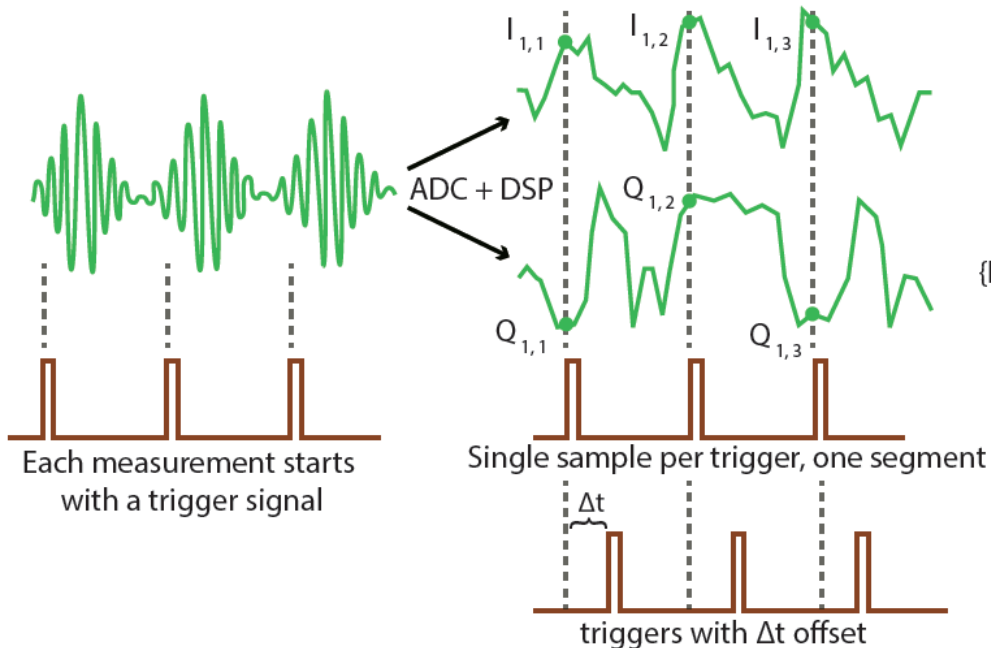
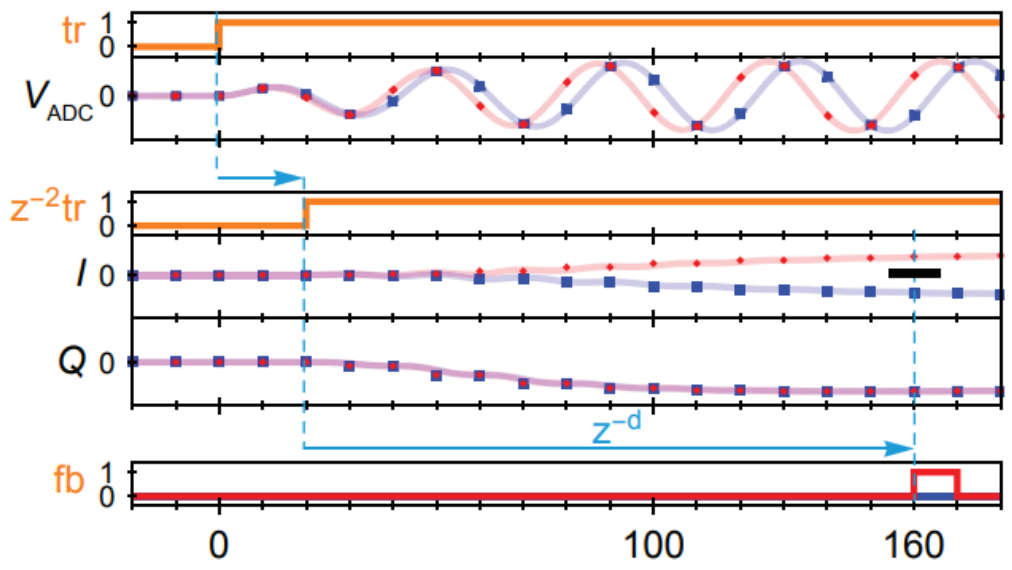
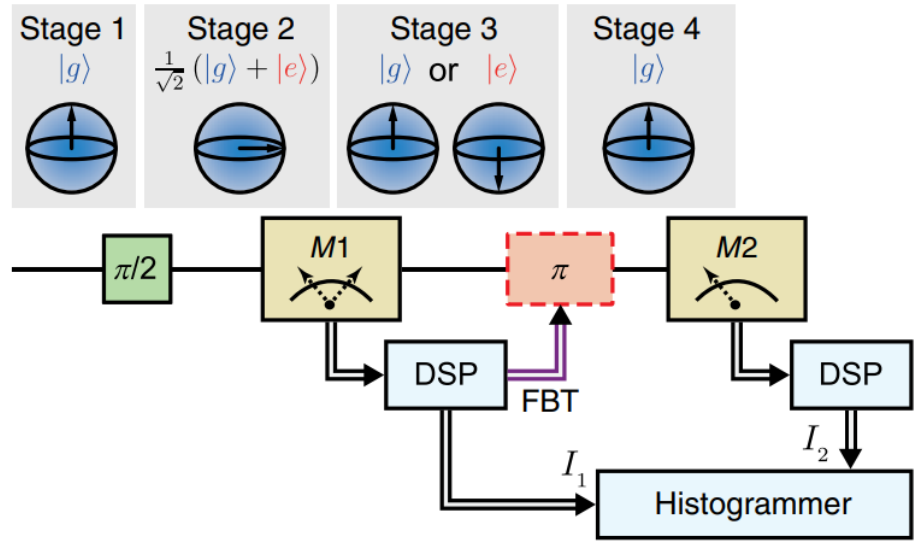


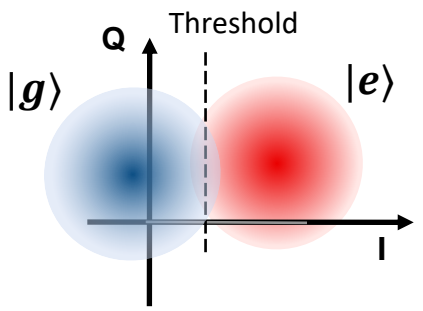
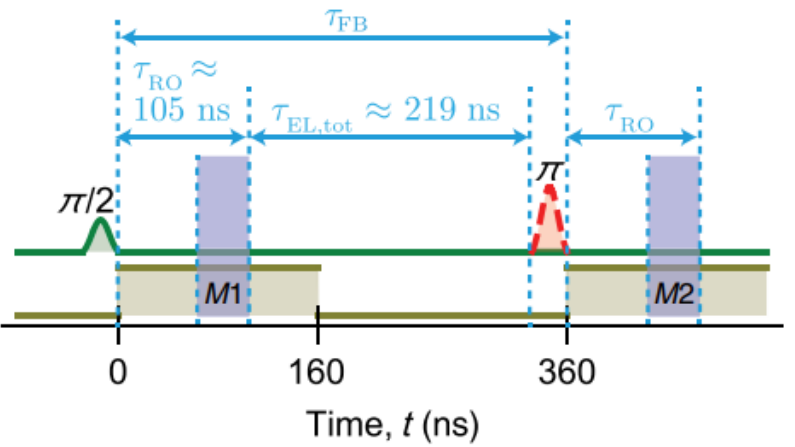
Figure 4. Readout fidelity measured for two qubits in the IBM Rochester quantum system (53 qubit processor). The

<https://www.ibm.com/blogs/research/2020/01/quantum-limited-amplifiers/>

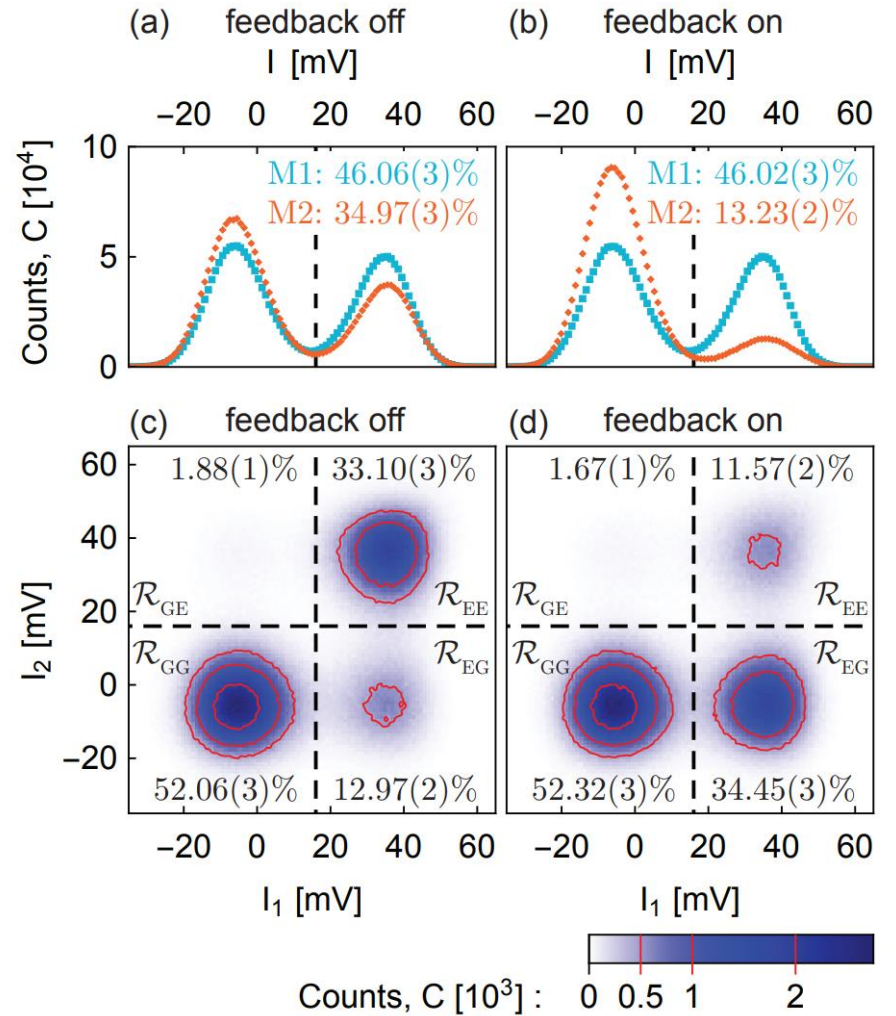
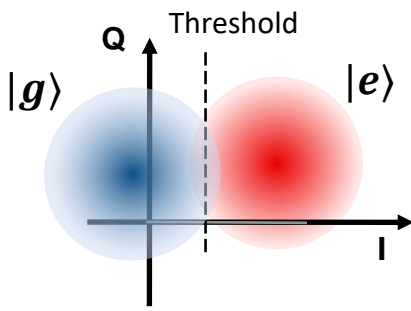
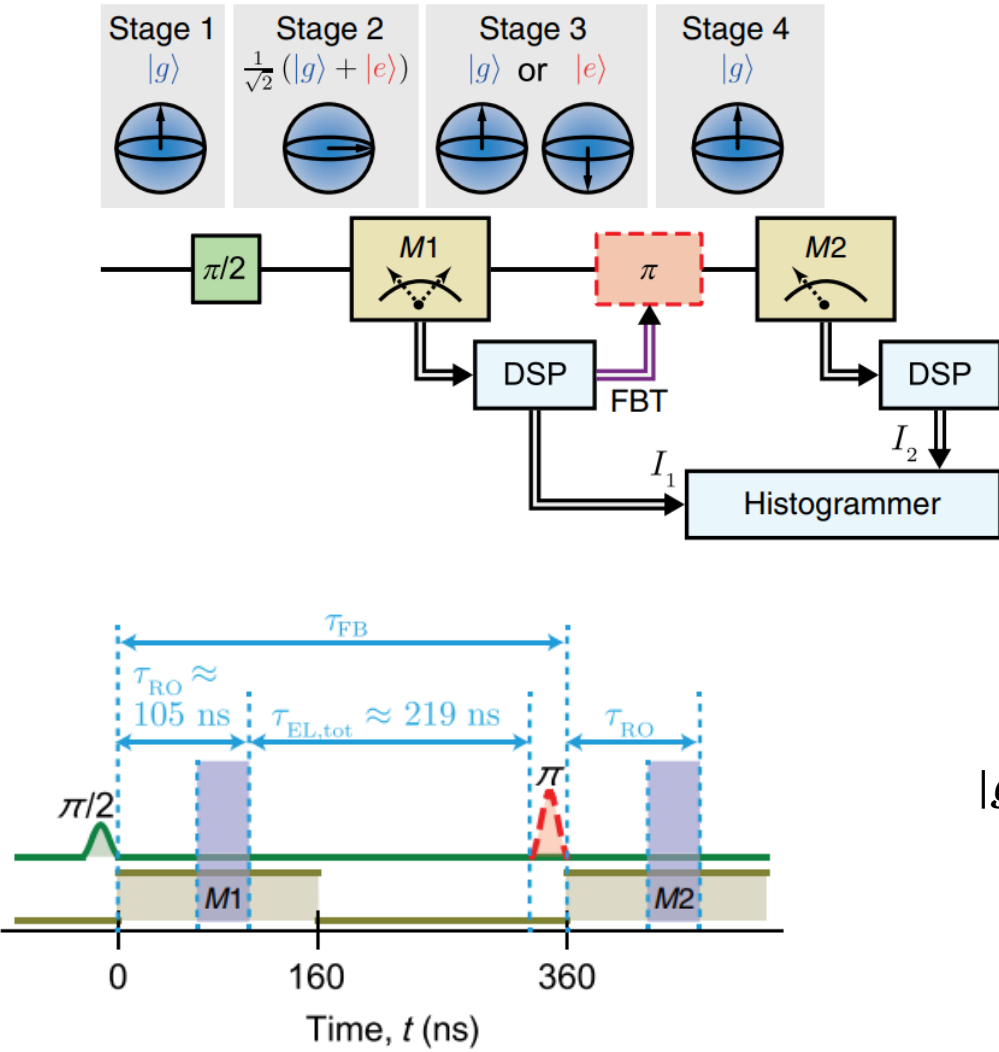
Utilization example of fast quantum state identification: Resetting qubit to $|g\rangle$ state



State identification and feedback generation



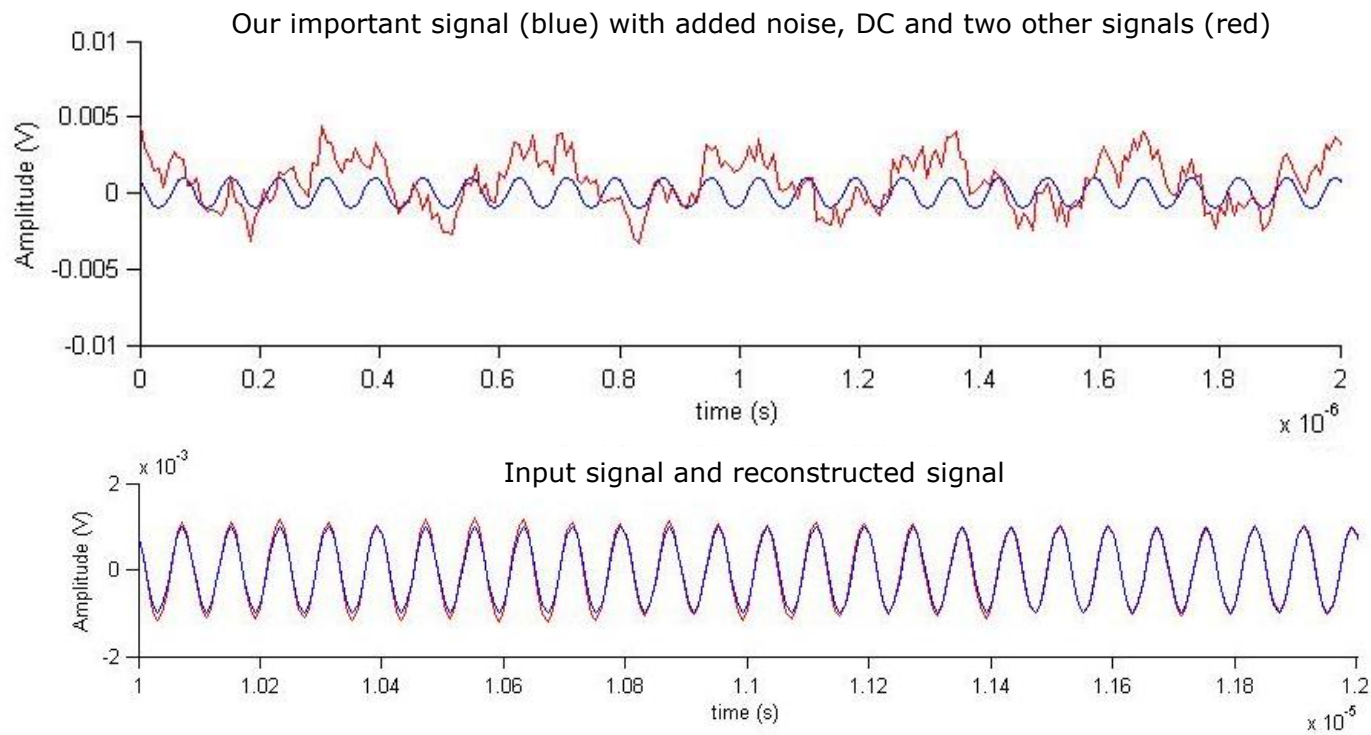
Utilization example of fast quantum state identification: Resetting qubit to $|g\rangle$ state



Lab Exercise: IQ Demodulation using MATLAB

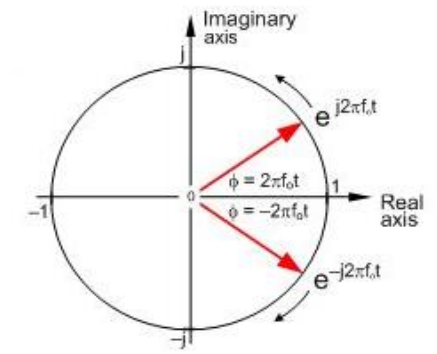
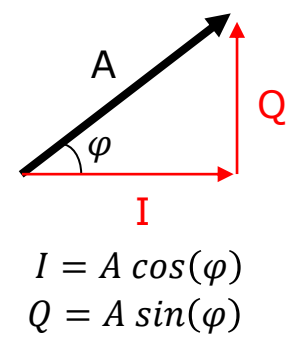
- Signals measured from quantum setup are often very noisy and mixed with different frequencies
- IQ demodulation is used to calculate the phase and amplitude of a signal to identify the quantum state

Lab Exercise

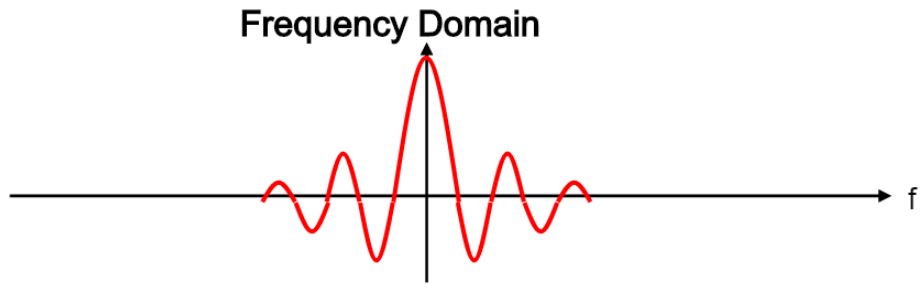
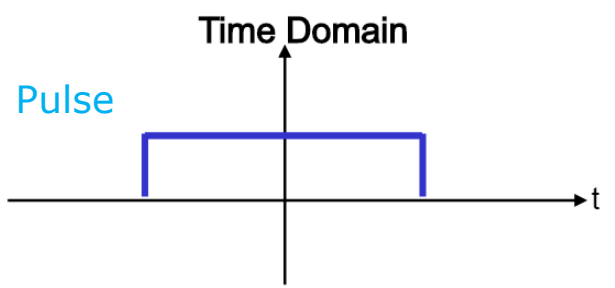
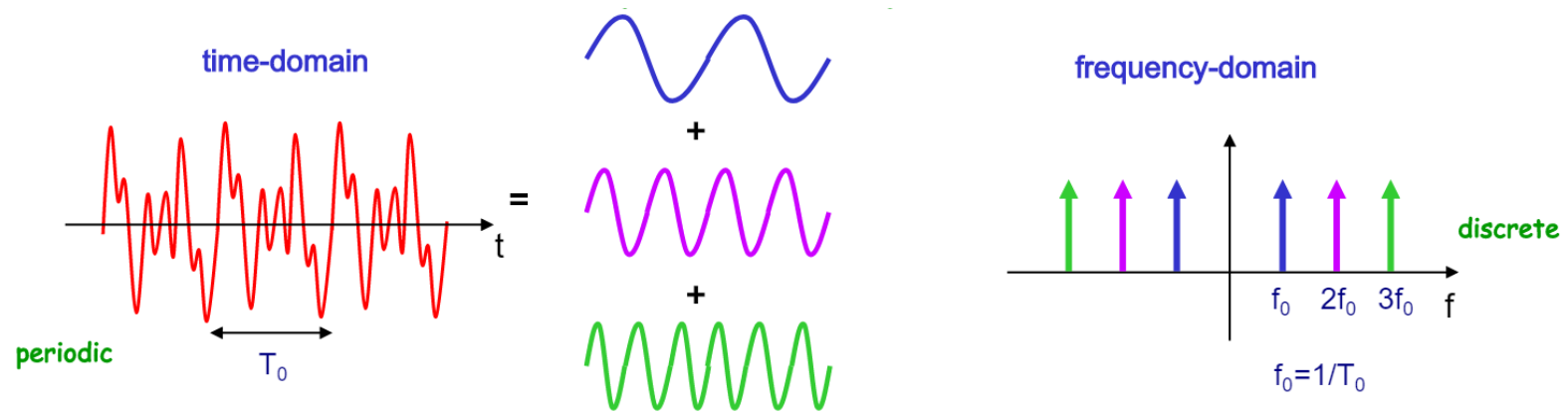


$$A e^{i\phi} = I + iQ$$

↑ signal amplitude ↑ Phase ↑ In-phase and quadrature components



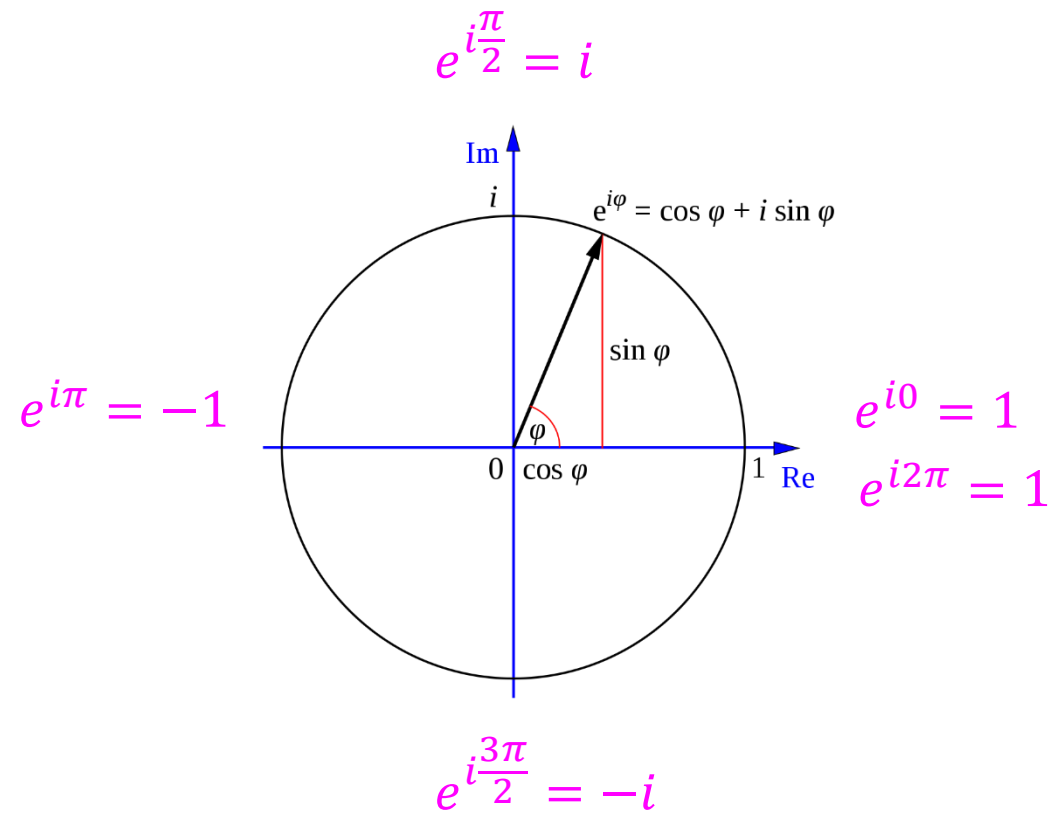
Any time domain signal can be represented as a sum of sinusoidal signals



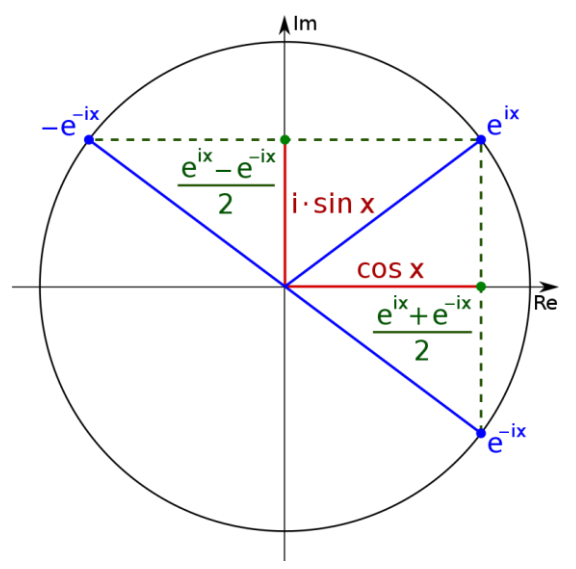
Euler's Formula

$$e^{ix} = \cos x + i \sin x$$

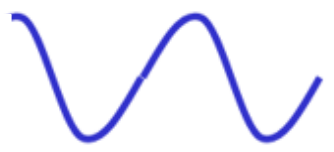
for any real number x



$$\cos x = \frac{e^{ix} + e^{-ix}}{2}$$



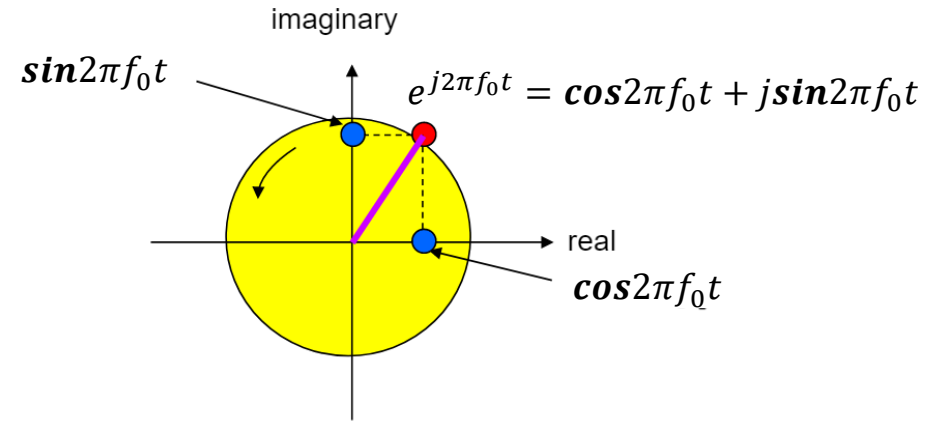
Time Domain



$$\cos(2\pi ft + \varphi)$$

$$= \frac{e^{i(2\pi ft + \varphi)}}{2} + \frac{e^{-i(2\pi ft + \varphi)}}{2}$$

Euler's Formula

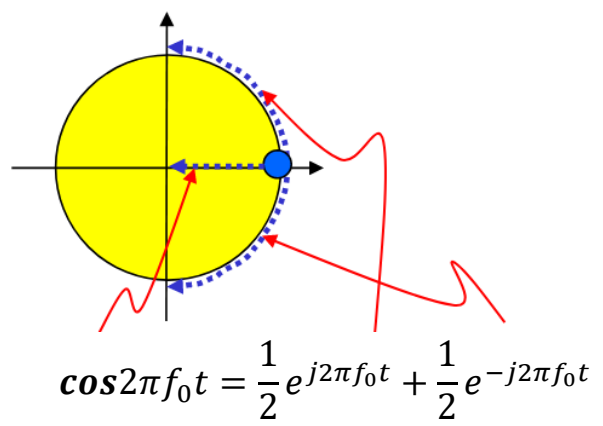
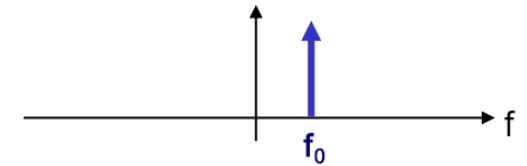


Time Domain

$$x(t) = e^{j2\pi f_0 t}$$

Frequency Domain

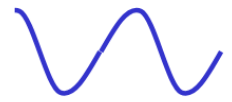
$$X(f) = \delta(f - f_0)$$



real signal in time positive frequency negative frequency

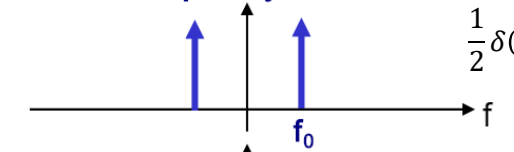
Time Domain

$$\cos(2\pi f_0 t)$$



Frequency Domain

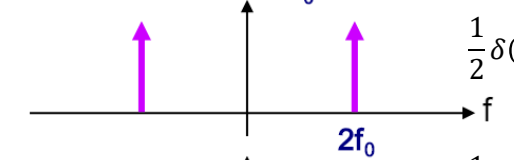
$$\frac{1}{2} \delta(f - f_0) + \frac{1}{2} \delta(f - f_0)$$



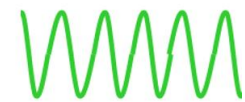
$$\cos(2\pi 2f_0 t)$$



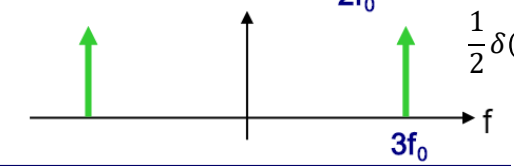
$$\frac{1}{2} \delta(f - 2f_0) + \frac{1}{2} \delta(f - 2f_0)$$



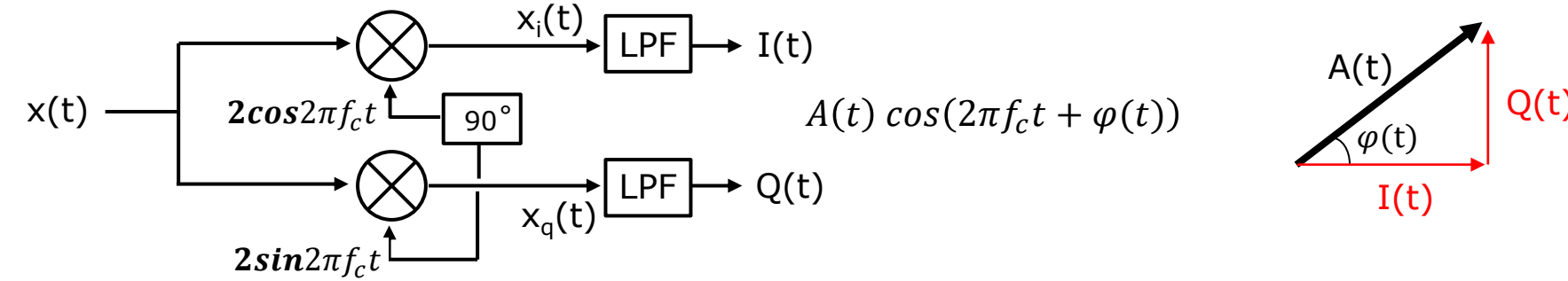
$$\cos(2\pi 3f_0 t)$$



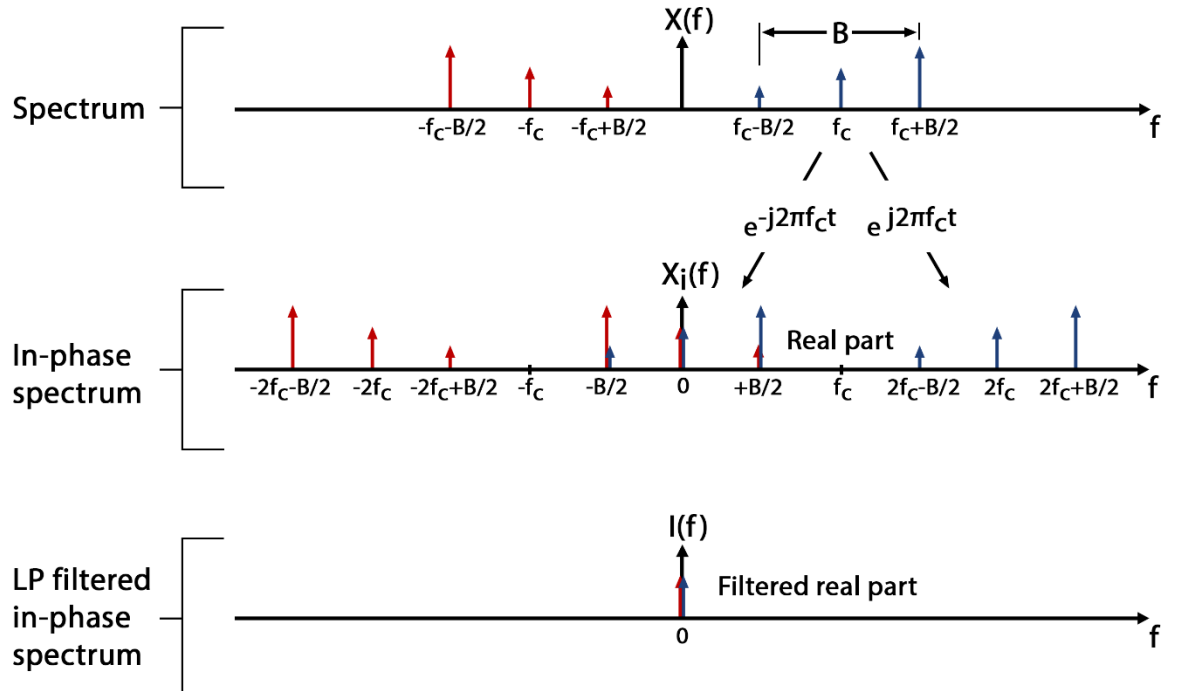
$$\frac{1}{2} \delta(f - 3f_0) + \frac{1}{2} \delta(f - 3f_0)$$



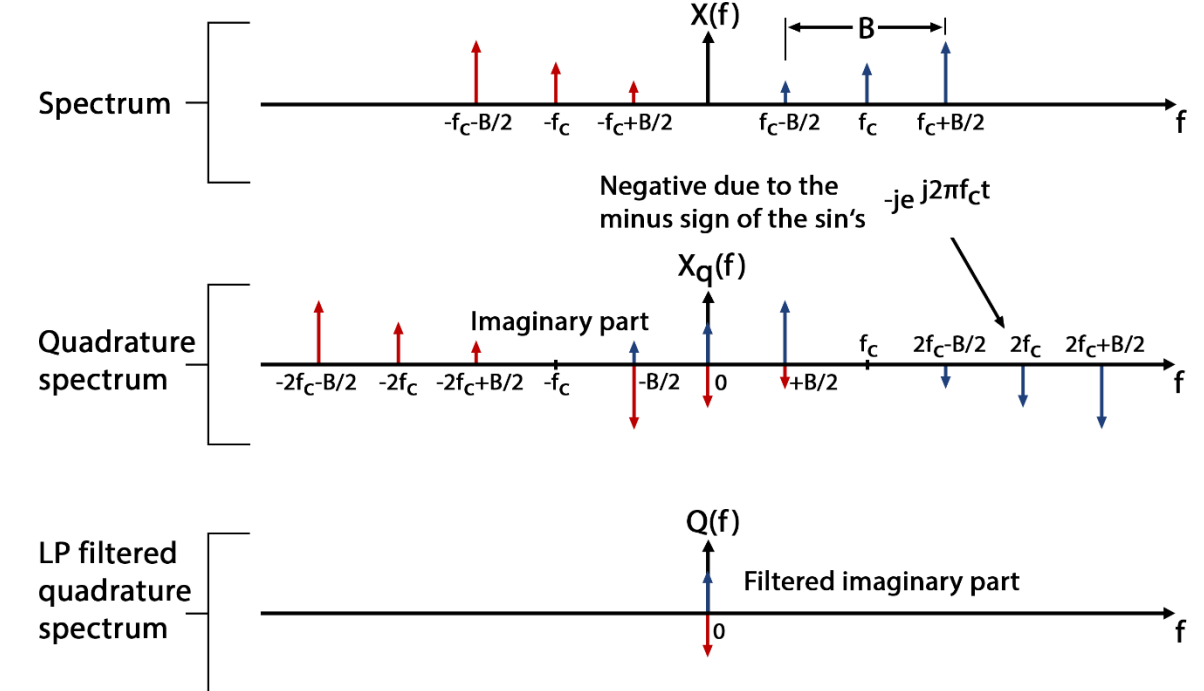
IQ Demodulation



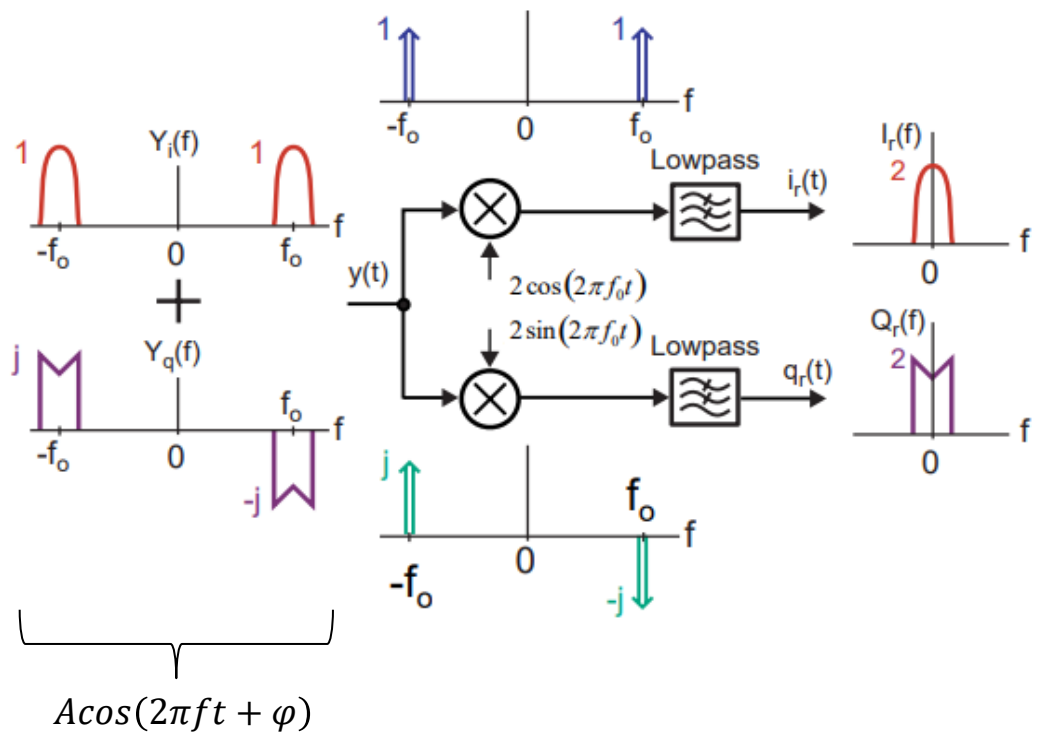
Multiplying Input Signal with Cos Signal



Multiplying Input Signal with Sin Signal



IQ Demodulation

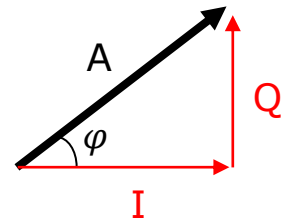


Important Trigonometric Identities:

$$\begin{aligned} \cos(a+b) &= \cos(a)\cos(b) - \sin(a)\sin(b) \\ \cos(a)\cos(a) &= 1/2(\cos(0) + \cos(2a)) \\ \sin(a)\cos(a) &= 1/2(\sin(2a) + \sin(0)) \\ \sin(a)\sin(a) &= 1/2(\cos(0) - \cos(2a)) \end{aligned}$$

$A\cos(2\pi ft + \varphi)$ is our signal that we want to reconstruct
 $A\cos(2\pi ft + \varphi) = A\cos(2\pi ft)\cos(\varphi) - A\sin(2\pi ft)\sin(\varphi)$

As known:
 $A\cos(\varphi) = I$
 $A\sin(\varphi) = Q$



$A\cos(2\pi ft + \varphi) = I\cos(2\pi ft) - Q\sin(2\pi ft)$ is our signal
 Lets multiply our signal with **LO**: $2 \times \cos(2\pi ft)$ and $2 \times \sin(2\pi ft)$

$$\begin{aligned} &2 \times A\cos(2\pi ft + \varphi) \times \cos(2\pi ft) \\ &= 2 \times (I\cos(2\pi ft) - Q\sin(2\pi ft)) \times \cos(2\pi ft) \\ &= I \times (\cos(0) + \cos(4\pi ft)) - Q \times (\sin(4\pi ft) + \sin(0)) \\ &= I + I \times \cos(4\pi ft) - Q \times \sin(4\pi ft) \end{aligned}$$

$$\begin{aligned} &2 \times A\cos(2\pi ft + \varphi) \times \sin(2\pi ft) \\ &= 2 \times (I\cos(2\pi ft) - Q\sin(2\pi ft)) \times \sin(2\pi ft) \\ &= I \times (\sin(4\pi ft) + \sin(0)) - Q \times (\cos(0) - \cos(4\pi ft)) \\ &= I \times \sin(4\pi ft) - Q + Q \times \cos(4\pi ft) \end{aligned}$$

If you have I and Q, you can calculate A and φ, and reconstruct your signal

$$A(t) = \sqrt{I(t)^2 + Q(t)^2} \qquad \varphi(t) = \tan^{-1} \frac{Q(t)}{I(t)}$$

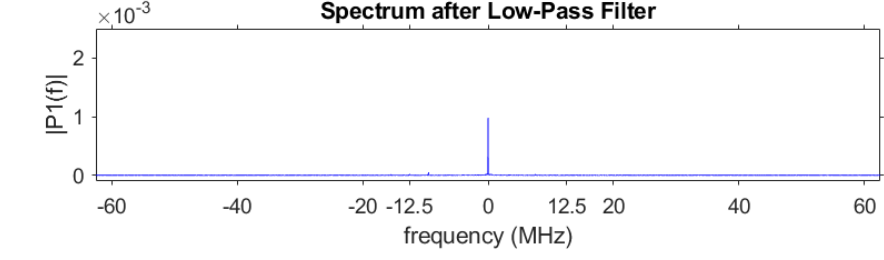
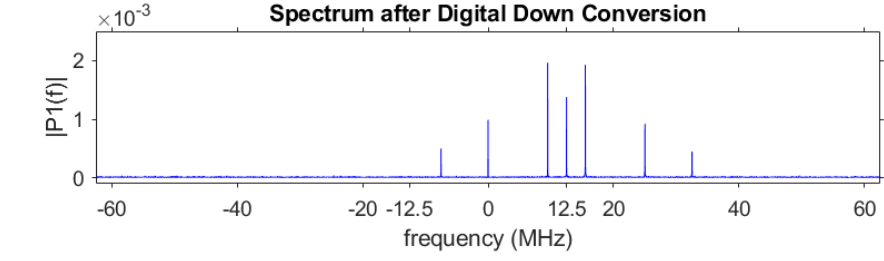
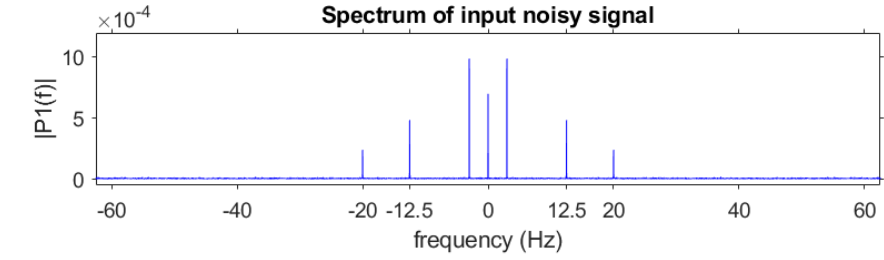
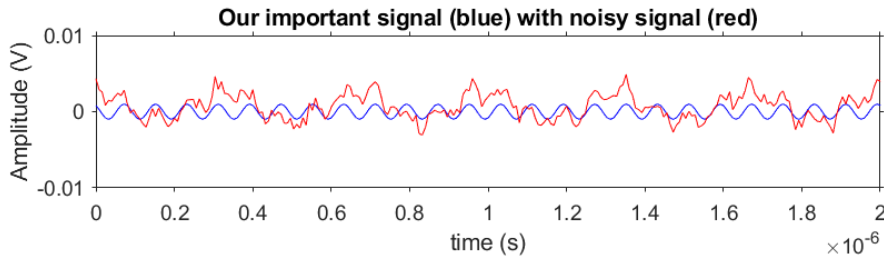
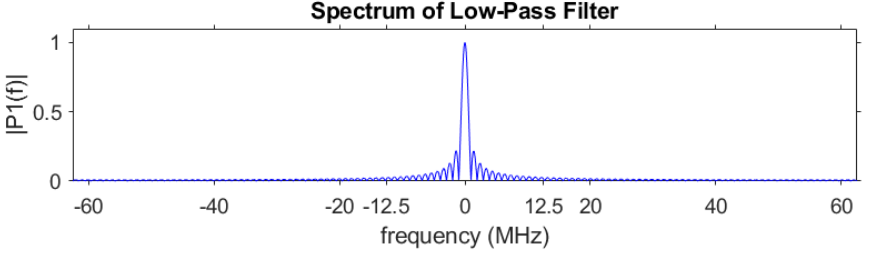
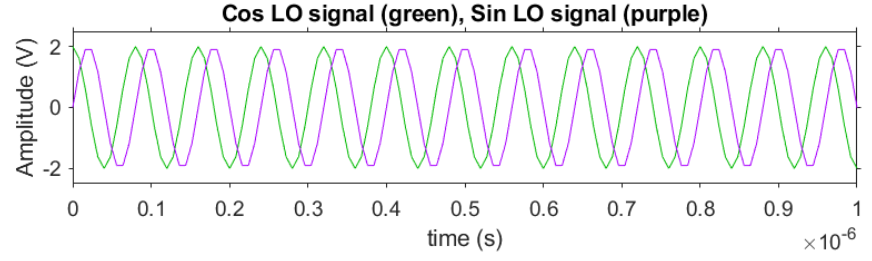
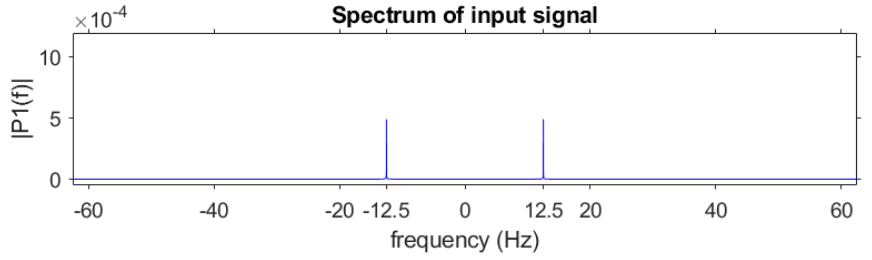
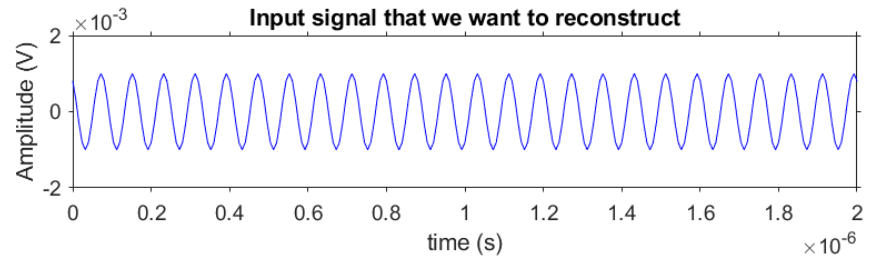
$$A(t) \cos(2\pi ft + \varphi(t))$$

Project Goal (Lockin_amplifier_solution.m)

Input Signal

LO

Low Pass Filter



Input Signal with Added Signals and noise

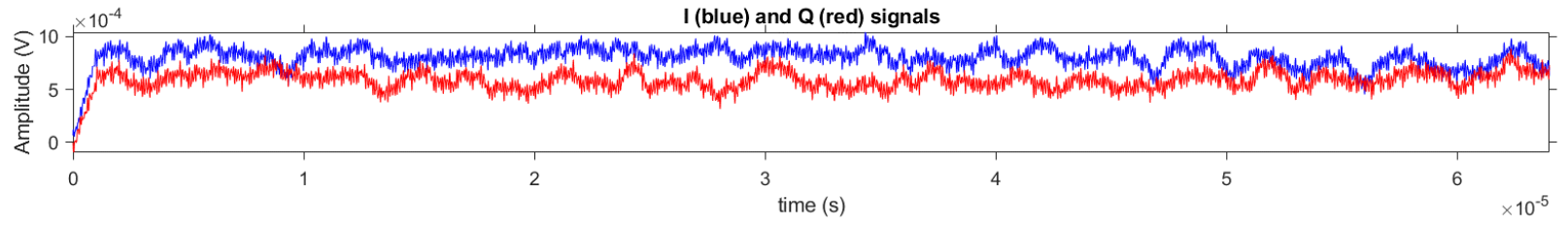
FFT

After DDC

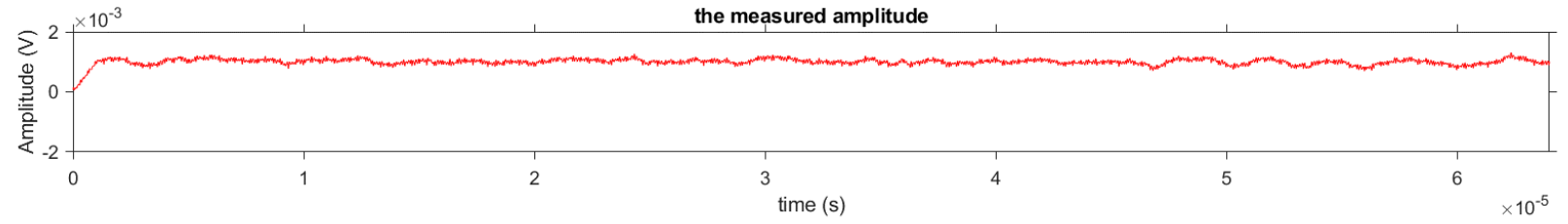
FFT of I & Q

Project Goal

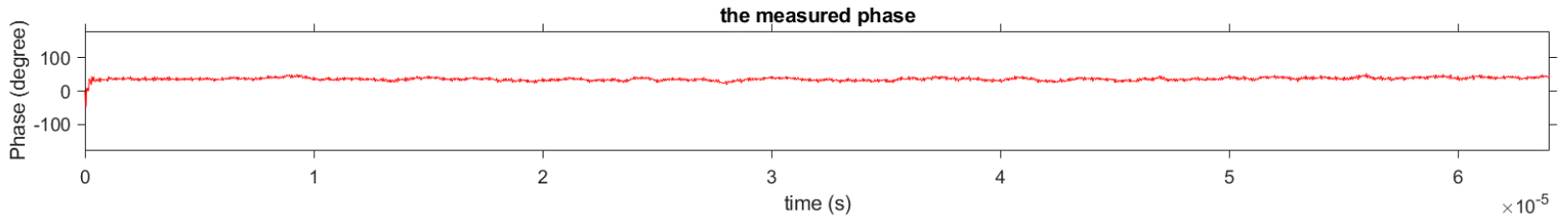
I & Q



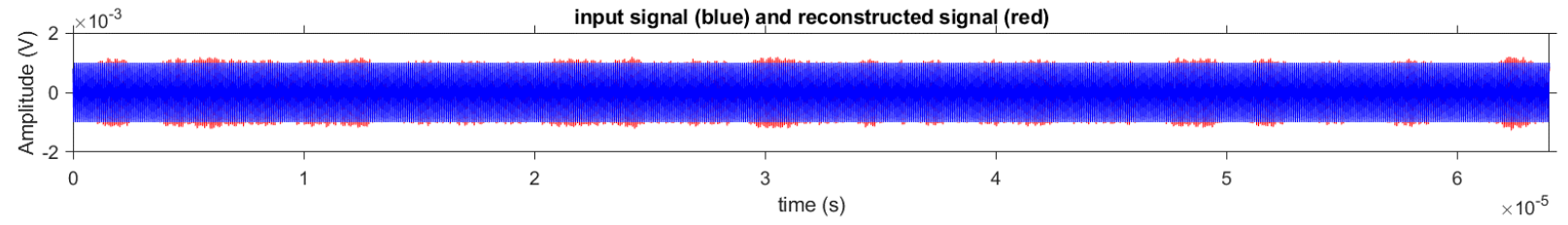
Amplitude



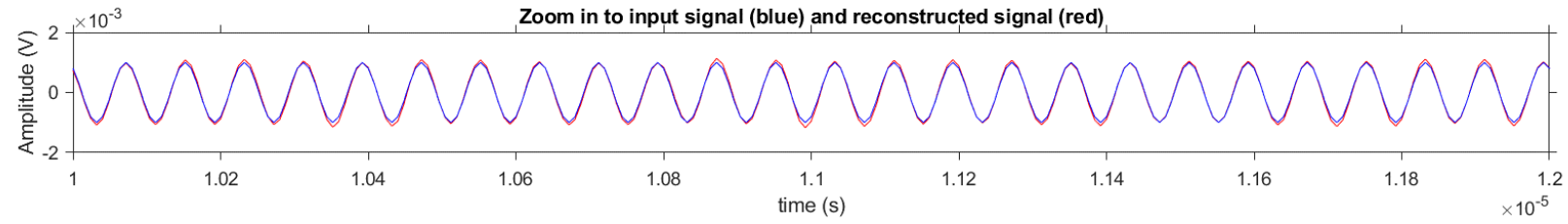
Phase



Reconstructed Signal



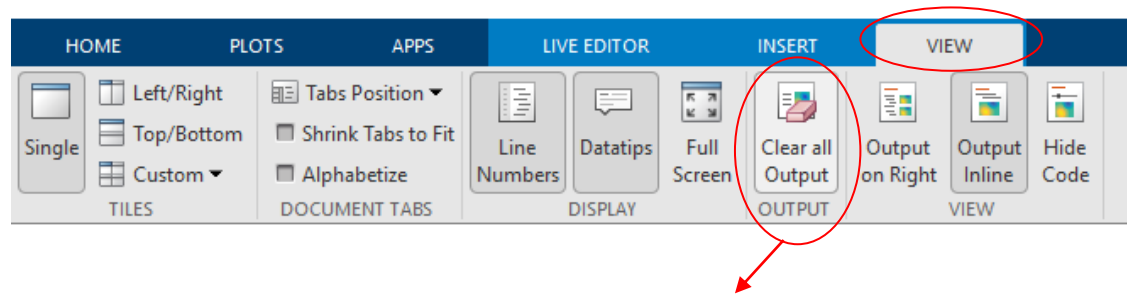
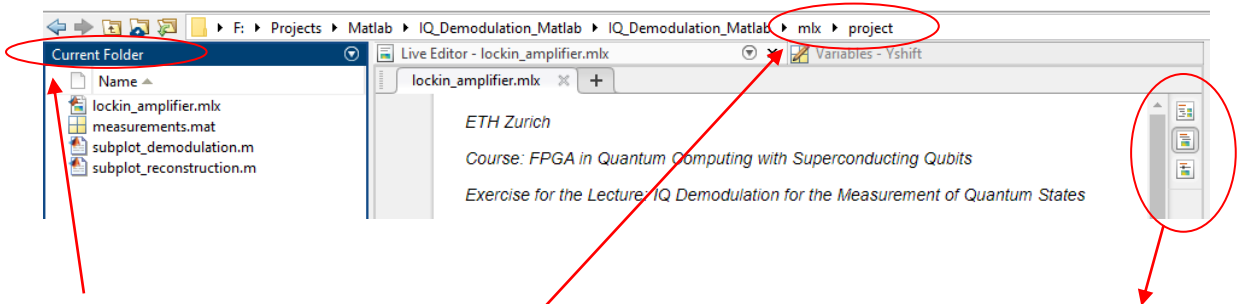
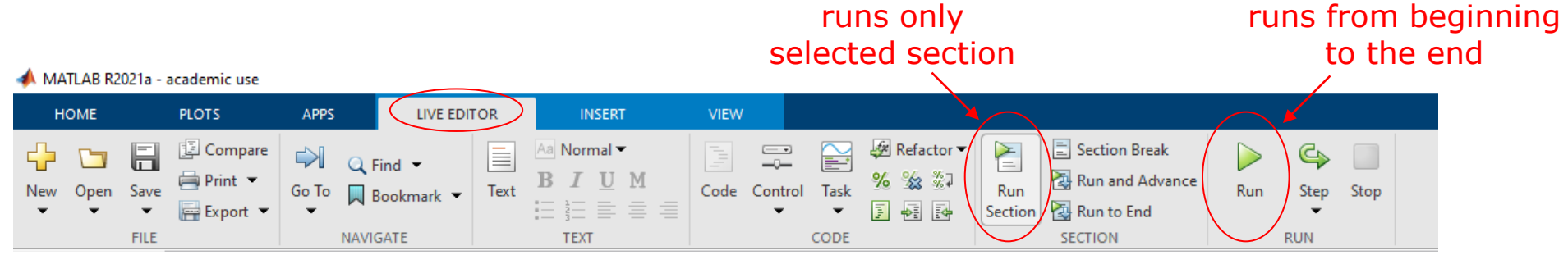
Reconstructed Signal (Zoomed)



This Week: IQ Demodulation to measure the phase and amplitude of a signal

Step by Step, Task 1, create the test signal:

- Open `lockin_amplifier.mlx` file in `IQ_Demodulation_Matlab\mlx\project` folder with MATLAB.
- You can use `lockin_amplifier.m` version for the MATLAB version older than R2016. Nevertheless we suggest you to use `.mlx` (live script). If you don't have new MATLAB installed, you may use MATLAB Online (<https://ch.mathworks.com/products/matlab-online.html>).
- Search `xxxxx` in the file to see what you need to add or change.
- Check how our input signal `signal_in` is generated (lines 1-12). What is the sampling period (in ns)? What is the phase of `signal_in` (in degree and radian) ?
- **Run Section** to see the signal.



Current Folder should show you only project files, but not subfolders

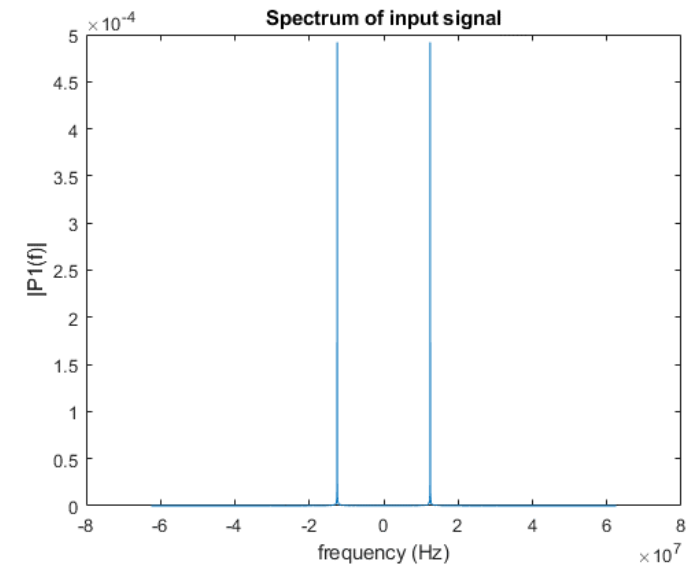
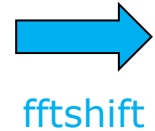
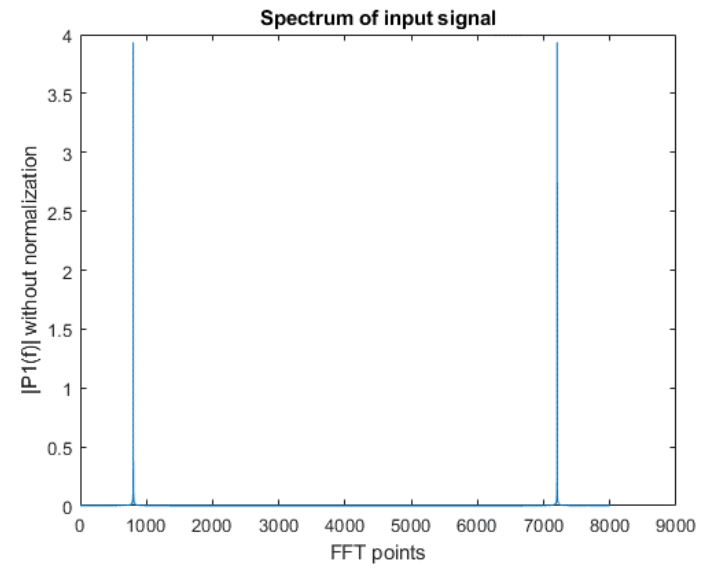
changes where to show output figures in your live script

clears all output figures

This Week: IQ Demodulation to measure the phase and amplitude of a signal

Step by Step, Task 1, create the test signal:

- Get the FFT of `signal_in` (line 25). **Run Section.**
- Generate `signal_s1` and `signal_s2`, which are added on top of `signal_in`. Use `t_range` to define time interval:
 - `signal_s1`: 3 MHz cos signal with 0.002 V amplitude, 0 radian phase (line 39)
 - `signal_s2`: 20 MHz sin signal with 0.0005 V amplitude, pi/4 radian phase (line 40)
- Get the FFT of `signal_in_noisy` (line 57)
- Apply `fftshift` (line 58). Normalize `fftshift` value with number of samples (check line 26 as an example). `fftshift` puts 0 Hz to the middle, otherwise 0 Hz is in the left edge, which is not conventional for FFT visualization. **Run Section.**



FFT result of 12.5 MHz cos without FFT shift

FFT result of 12.5 MHz cos signal after FFT shift and normalization

This Week: IQ Demodulation to measure the phase and amplitude of a signal

Step by Step, Task 3, reconstruct the signal:

- Now we have I and Q components. We can extract amplitude and phase from I and Q. Then we can reconstruct our signal. *A_reconstruct* is the amplitude, *P_reconstruct* is the phase. Calculate them using the formulas below. Use `atan2(y,x)` function of matlab to calculate the arctangent. (Line 136-137)

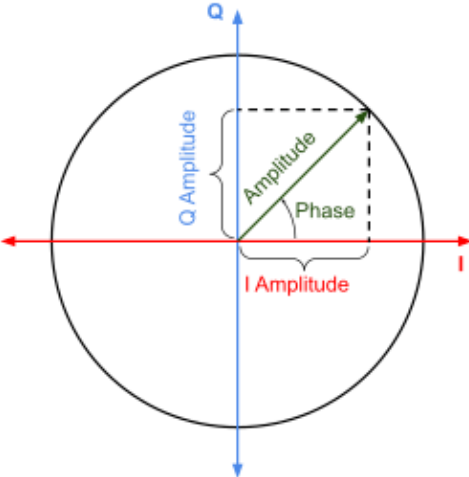
$$A(t) = \sqrt{I(t)^2 + Q(t)^2} \qquad \varphi(t) = \tan^{-1} \frac{Q(t)}{I(t)}$$

- Reconstruct the signal using the formula below. (Line 139)

$$A(t) \cos(2\pi ft + \varphi(t))$$

➤ **Run Section.**

- Is the reconstruction successful ? Check the phase and amplitude plots. Is it matching with your input signal's amplitude and phase ?
- Change the phase of your input signal *signal_in* between 0 and 2π , and check if the reconstruction is correct.
- Check the plots of I and Q values and the Phase. Considering that I and Q values are positive or negative, phase can be at one of the four regions around the sphere (0 to 2π). Does it match in your plot (for example when both I and Q are negative, phase should be between π and 1.5π)?



This Week: IQ Demodulation to measure the phase and amplitude of a signal

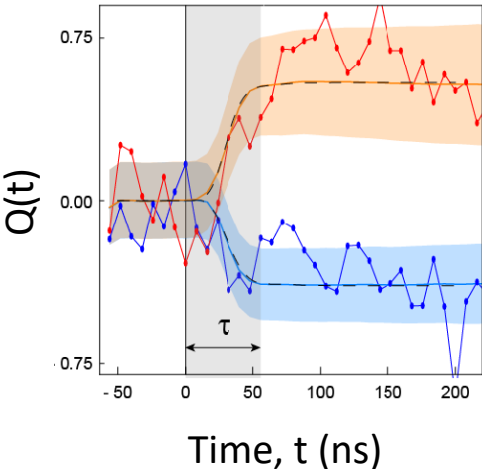
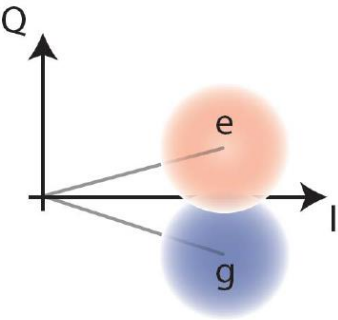
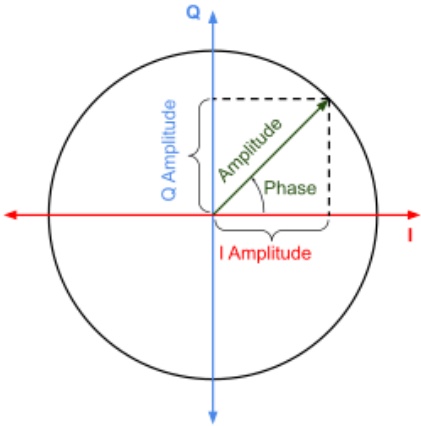
Step by Step, Task 3, state identification:

- By looking at I and Q components and their region at complex plane, we can distinguish two qubit states (e and g) after measurement.
- `measurements.mat` file includes 3 measurement signals. Type `clear all` in console. Click `measurements.mat`. You will see 3 data sequence in workspace. In order to automatize opening the signals, you can write `load('measurements.mat')` at the very beginning of your code.
- You expect to see your quantum signal at 12.5 MHz after analog down conversions. Since your IQ modulation for the signal reconstruction also uses 12.5 MHz LO signal, you can use the same code for the state identification just by changing the input.
- In `line 13`, remove comment out for ignoring the input signal that you created: `signal_in = zeros(1,samples);`
- In `line 42`, remove comment out to process quantum measurement result: `signal_in_noisy = measurement_1;`
- Run all code, and repeat the runs by assigning `measurement_2` and `measurement_3` to `signal_in_noisy`.
- Answer the questions for each run, considering that you expect a response from your qubits as given in figures below:

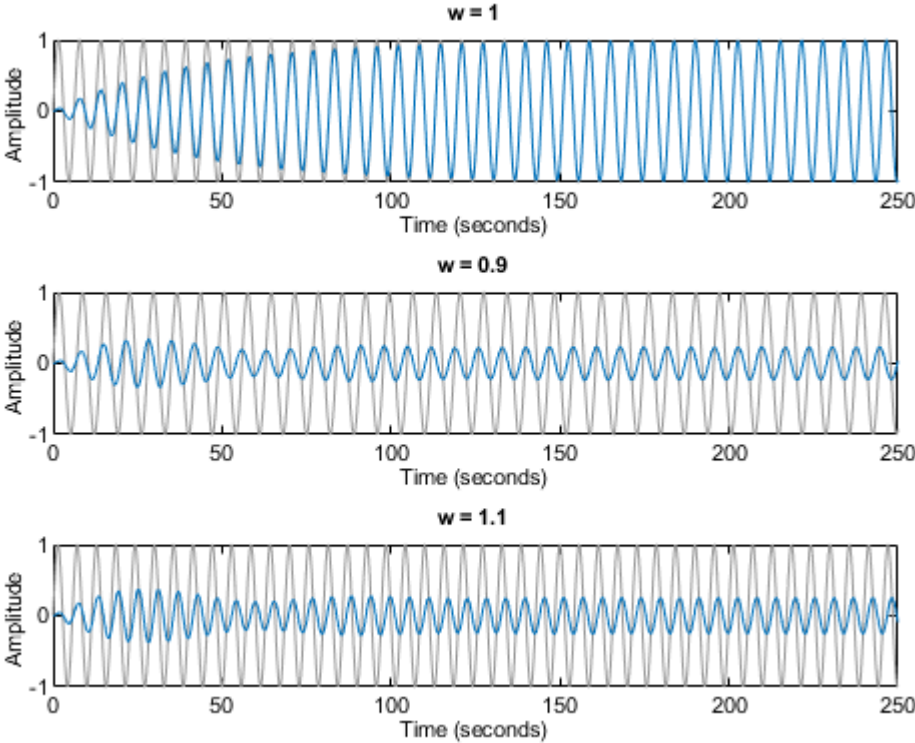
When do the measurements start and stop ?

What is the qubit's state for signal 1 & 2 ?

What happens at $2.8e-5$ s in signal 3 ?



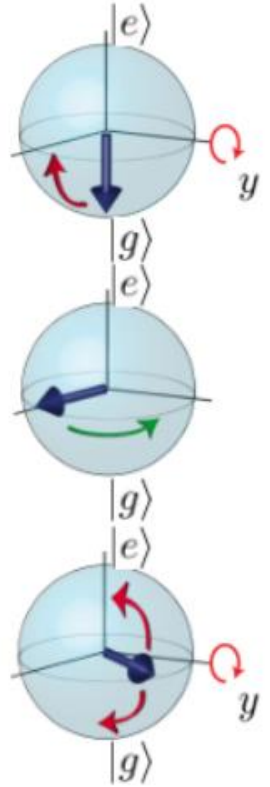
Additional Sources and References



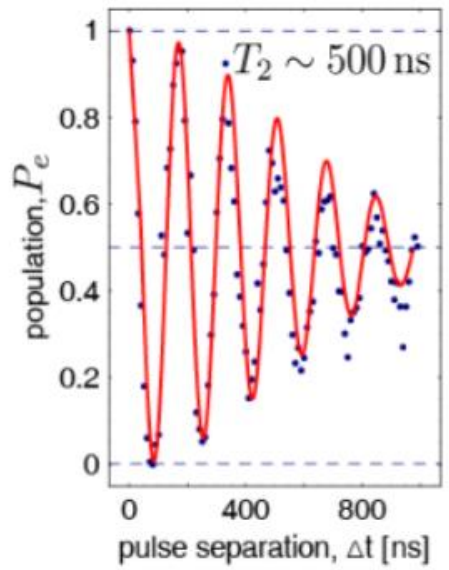
<https://ch.mathworks.com/help/control/ug/analyzing-the-response-of-an-rlc-circuit.html>

RLC resonance

pulse scheme:



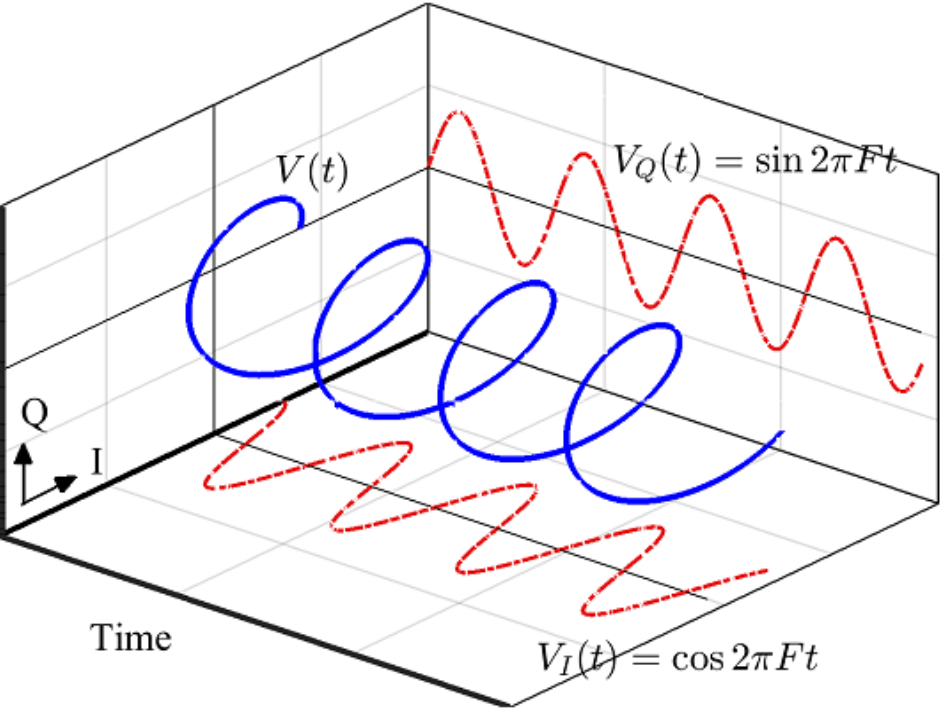
Ramsey fringes:



<https://qudev.phys.ethz.ch/static/content/QSIT14/QSITNotes.pdf>
<https://arxiv.org/pdf/1312.1463.pdf>;
[http://www.cs.virginia.edu/~robins/The Limits of Quantum Computers.pdf](http://www.cs.virginia.edu/~robins/The_Limits_of_Quantum_Computers.pdf)

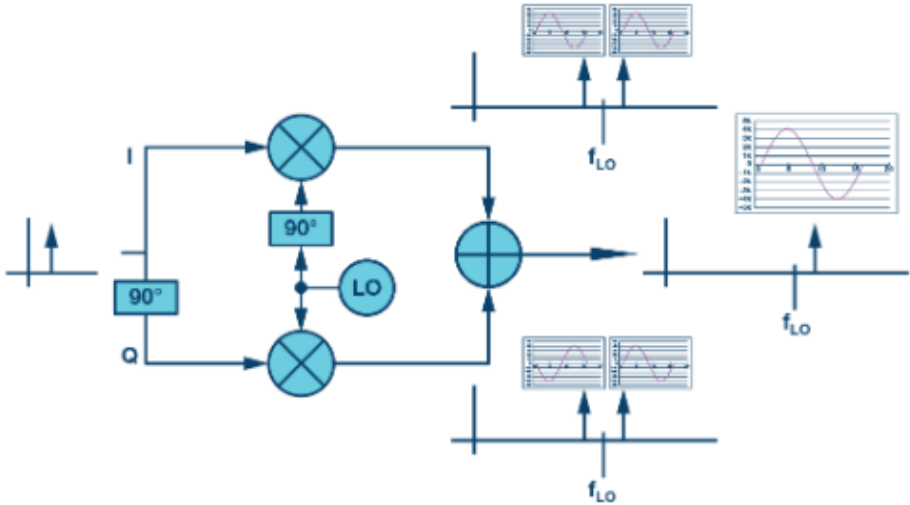
Introduction to Quantum Science

Additional Sources and References



<https://www.dsprelated.com/showarticle/192.php>
<https://wirelesspi.com/the-concept-of-frequency/>
<https://slideplayer.com/slide/10939951/>
https://en.wikipedia.org/wiki/Euler%27s_formula

IQ modulation and demodulation



<https://www.analog.com/ru/analog-dialogue/articles/complex-mixers-zif-architecture-advanced-algorithms-black-magic-next-generation-sdr-transceivers.html>

RF Mixers

Additional Sources and References

Verilog In One Day

Feb-9-2014

Verification IP's

SystemVerilog, Vera, E, SystemC, Verilog

ASIC WORLD Directory

- Verilog
- Tutorial
- Examples
- Questions
- Tools
- Books
- Links
- FAQ

- Sponsor
- Home
- Disclaimer
- FAQ

Hot Jobs

- **Introduction**
 - Block diagram of arbiter
 - Low level design
- **Modules**
 - Code of module "arbiter"
- **Data Type**
- **Operators**
- **Control Statements**
 - If-else
 - Case
 - While
 - For loop
 - Repeat
 - Summary
- **Variable Assignment**

<https://www.nandland.com/verilog/examples/>
<http://www.asic-world.com/verilog/veritut.html>

Coding Verilog

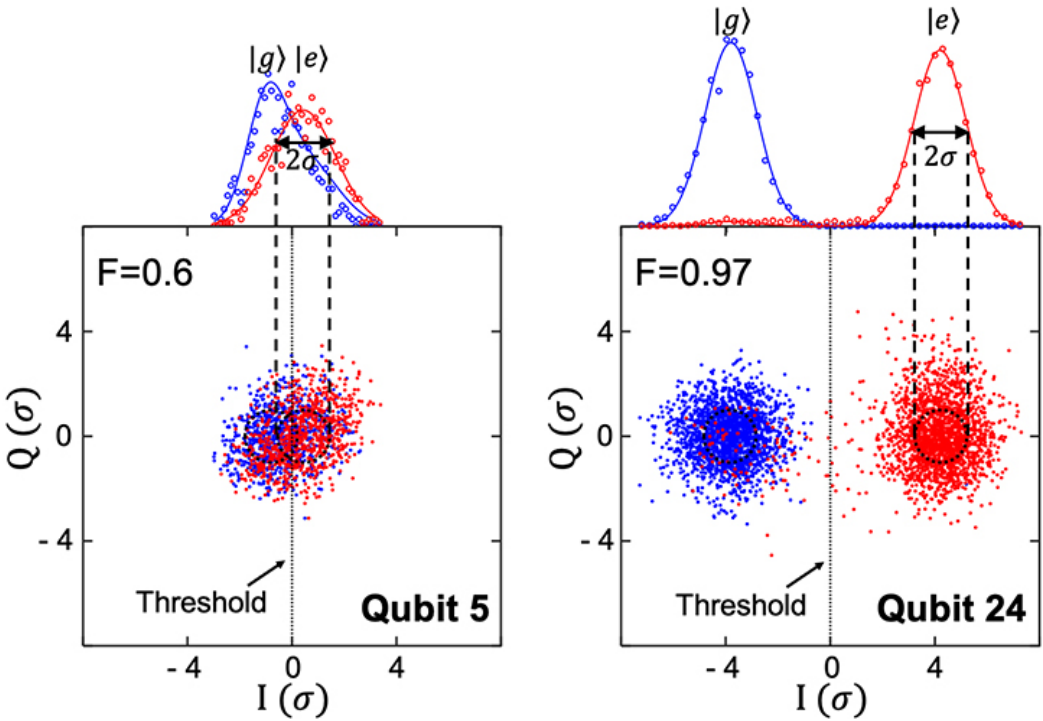


Figure 4. Readout fidelity measured for two qubits in the IBM Rochester quantum system (53 qubit processor). The

https://en.wikipedia.org/wiki/Fidelity_of_quantum_states
<https://www.ibm.com/blogs/research/2020/01/quantum-limited-amplifiers/>

Quantum Fidelity

Special Thanks



Prof. Andreas Wallraff
Head of QEC and Quantum Device Lab (Qudev)
Head of Institute for Solid State Physics

References

- K. Reuer., "Real time, single shot, dispersive readout of Superconducting qubits using a field Programmable gate Array", Master Thesis, ETH (2018)
- N. Schwegler, "Towards Low-Latency Parallel Readout of Multiple Trapped Ions", Master Thesis, ETH (2018)
- Y. Salathe, et al., "Low-Latency Digital Signal Processing for Feedback and Feedforward in Quantum Computing and Communication", Phys. Rev. Applied (2018)
- T. Walter, et al., "Rapid, High-Fidelity, Single-Shot Dispersive Readout of Superconducting Qubits" Phys. Rev. Applied 7 (2017)
- A. Wallraff, "Quantum Information Processing", ETH Lecture (2018)
- C. Eichler, "Quantum Science with Superconducting Qubits", ETH Lecture (2019)
- P. Kurpiers, et al., "Quantum Communication with Time-Bin Encoded Microwave Photons", Phys. Rev. Applied (2019)
- P. Kurpiers, et al. "Deterministic quantum state transfer and remote entanglement using microwave photons", Nature 558, 264-267 (2018)