Introduction to the projects

SLAC Summer Institute

August 2026

Chelsea Bartram, Jenny Smith, Zoe Smith, Caitlyn Stone-Whitehead

Three projects over the course of the week!

- Project #1:

Develop a better understanding of Josephson junctions as a building block of quantum circuits

Use simulation software to build quantum circuits and understand their properties

- Project #2:

Simulating phonon propagation and quasiparticle response, and analyzing the results using G4CMP.

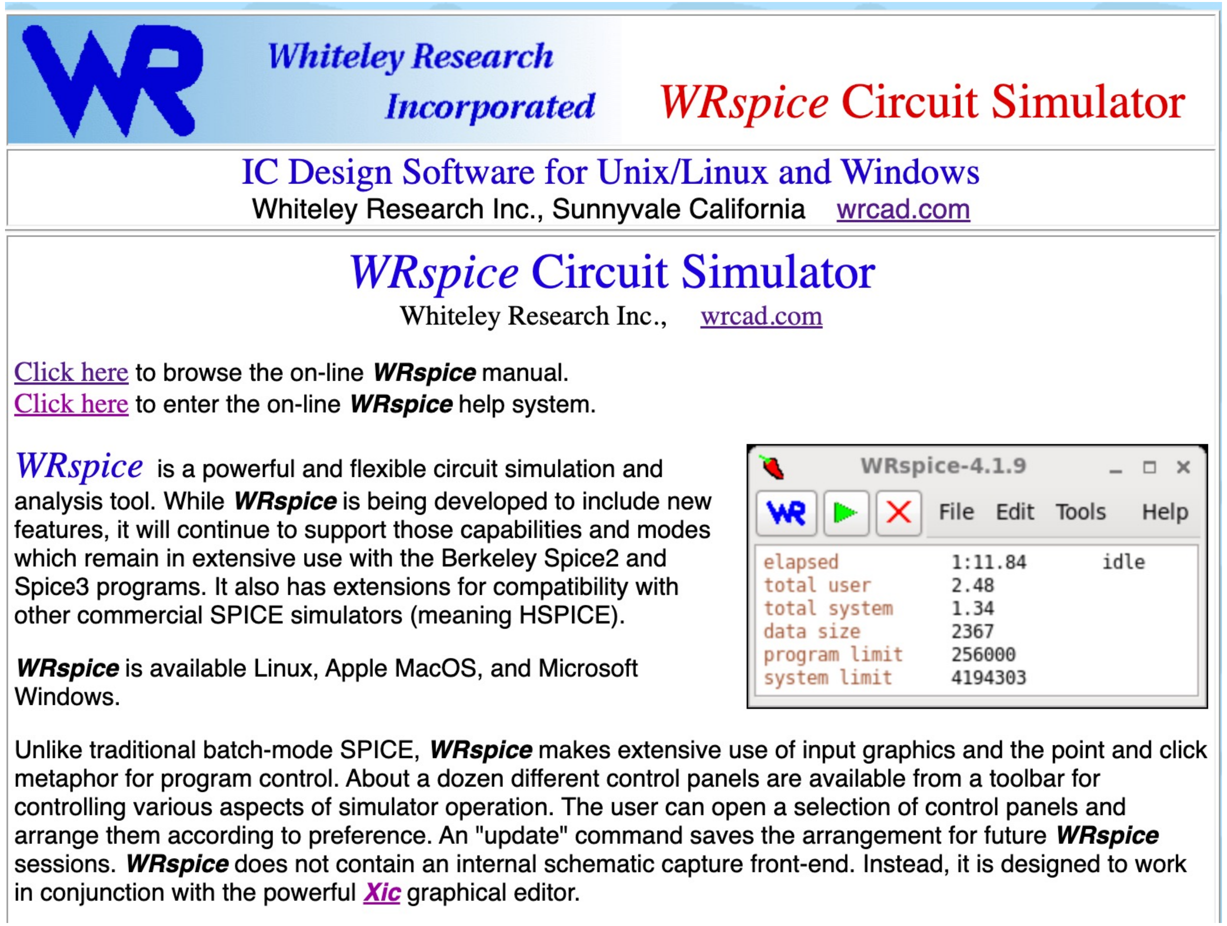
- Project #3:

SMuRF demo, mult-device measurements using real fridge measurements

Project software

Project 1: WRSPICE Tutorial

- Used for time domain or frequency simulations of quantum circuits
- Specific capability to simulate Josephson junctions
- Similar to SPICE
- Good at simulating continuous wave amplifiers…SQUIDs, JPAs, TWPAs, etc



Whiteley Research Incorporated *WRspice* Circuit Simulator

IC Design Software for Unix/Linux and Windows
Whiteley Research Inc., Sunnyvale California wrcad.com

WRspice Circuit Simulator

Whiteley Research Inc., wrcad.com

Click here to browse the on-line ***WRspice*** manual.
Click here to enter the on-line ***WRspice*** help system.

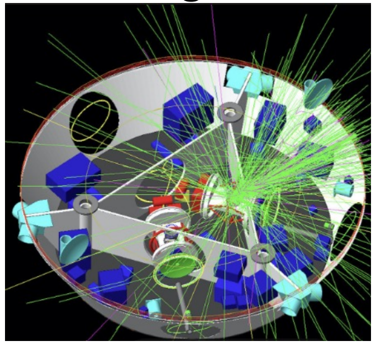
WRspice is a powerful and flexible circuit simulation and analysis tool. While ***WRspice*** is being developed to include new features, it will continue to support those capabilities and modes which remain in extensive use with the Berkeley Spice2 and Spice3 programs. It also has extensions for compatibility with other commercial SPICE simulators (meaning HSPICE).

WRspice is available Linux, Apple MacOS, and Microsoft Windows.

Unlike traditional batch-mode SPICE, ***WRspice*** makes extensive use of input graphics and the point and click metaphor for program control. About a dozen different control panels are available from a toolbar for controlling various aspects of simulator operation. The user can open a selection of control panels and arrange them according to preference. An "update" command saves the arrangement for future ***WRspice*** sessions. ***WRspice*** does not contain an internal schematic capture front-end. Instead, it is designed to work in conjunction with the powerful ***Xic*** graphical editor.

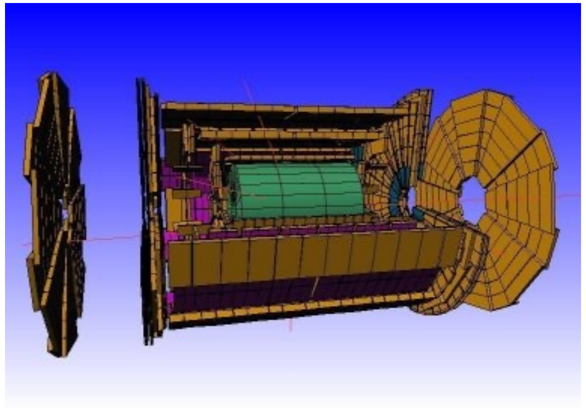
Project 2: G4CMP Tutorial

Shielding Studies



Picture Gallery | geant4.web.cern.ch

ATLAS



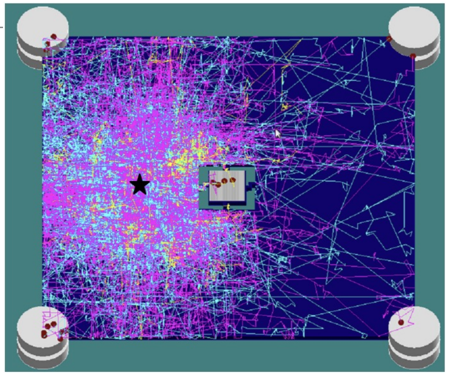
Picture Gallery | geant4.web.cern.ch

GEANT4 A SIMULATION TOOLKIT + G4CMP

- The reference C++ toolkit for HEP
- Handles particle interactions with matter
- Highly customizable for different geometries, sources and processes

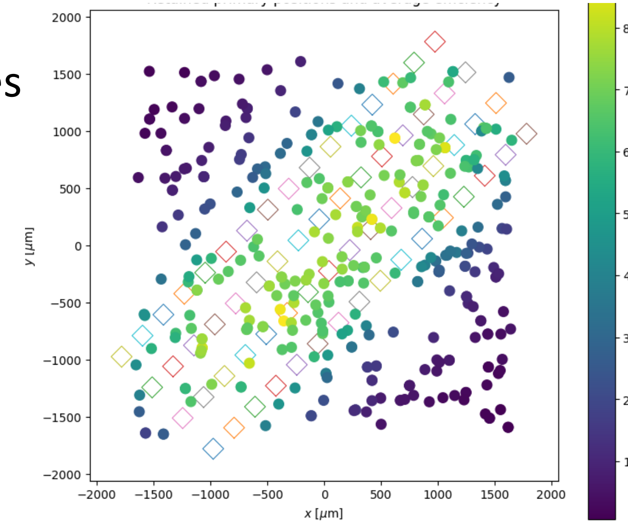
- Energy deposition excites phonons and charges OR phonons and quasiparticles
- Phonons/QP/charges applicable processes
- Applicable to < 1K temperatures

Phonon Mitigation



[G4CMP: Condensed Matter Physics Simulation Using the Geant4 Toolkit]

Collection Efficiency



- [Recent Developments in Geant4], J. Allison et al., Nucl. Instrum. Meth. A 835 (2016) 186-225
- [Geant4 Developments and Applications], J. Allison et al., IEEE Trans. Nucl. Sci. 53 (2006) 270-278
- [Geant4 - A Simulation Toolkit], S. Agostinelli et al., Nucl. Instrum. Meth. A 506 (2003) 250-303

Project 3: SMURF

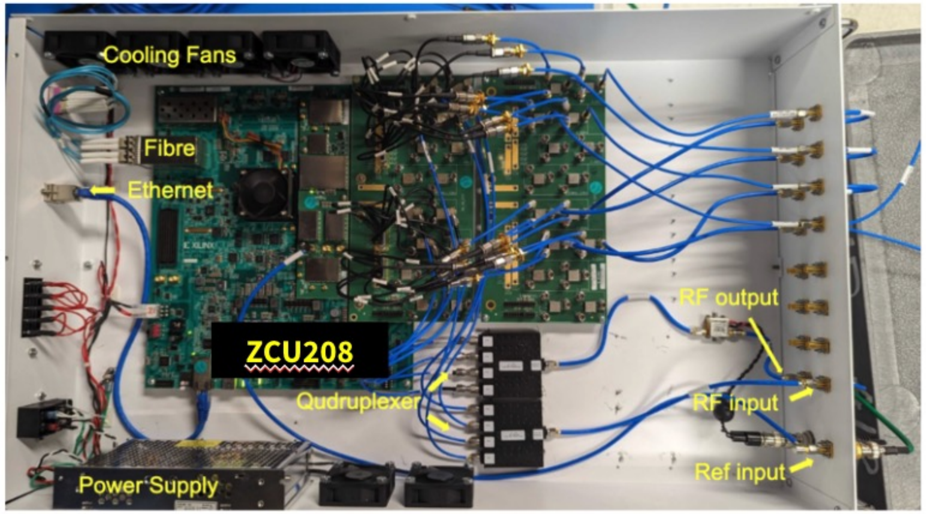
SLAC FPGA-Based Readout

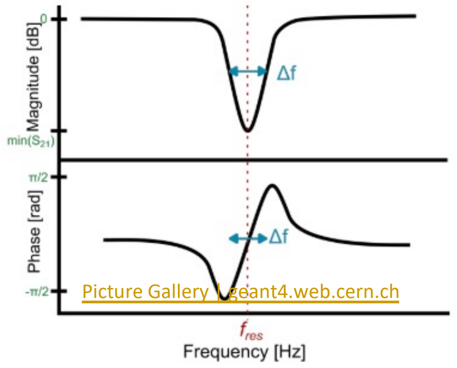
SLAC Microresonator RF (SMuRF) Electronics: A tone-tracking readout system for superconducting microwave resonator arrays

Cyndia Yu,[1,2,a)] Zeeshan Ahmed,[2,3] Josef C. Frisch,[3] Shawn W. Henderson,[2,3] Max Silva-Feaver,[4] Kam Arnold,[4] David Brown,[3] Jake Connors,[5] Ari J. Cukierman,[1,2] J. Mitch D'Ewart,[3] Bradley J. Dober,[5] John E. Dusatko,[3] Gunther Haller,[3] Ryan Herbst,[3] Gene C. Hilton,[5] Johannes Hubmayr,[5] Kent D. Irwin,[1,3] Chao-Lin Kuo,[1,3] John A.B. Mates,[5] Larry Ruckman,[3] Joel Ullom,[5] Leila Vale,[5] Daniel D. Van Winkle,[3] Jesus Vasquez,[3] and Edward Young[1,2]

[1)] *Department of Physics, Stanford University; Stanford, CA 94305; USA*
[2)] *Kavli Institute for Particle Astrophysics and Cosmology; Stanford, CA 94305; USA*
[3)] *SLAC National Accelerator Laboratory; Menlo Park, CA 94025; USA*
[4)] *Department of Physics, University of California San Diego; La Jolla, CA 92093; USA*
[5)] *National Institute of Standards and Technology; Boulder, CO 80305; USA*

(Dated: 24 August 2022)





Picture Gallery | geant4.web.cern.ch

FIG. 1. Example S_{21} amplitude [Top] and phase [Bottom] versus frequency for an idealized resonator. The resonance frequency f_{res} corresponds here to the minimum of transmission. The width characterizes the sharpness of the resonance, which is inversely related to the quality factor Q.

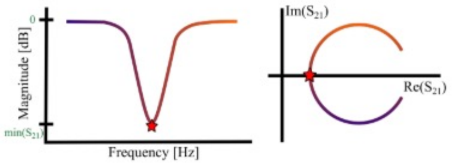
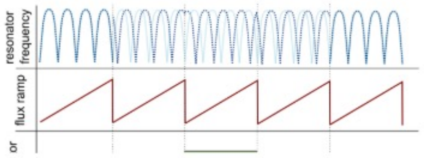
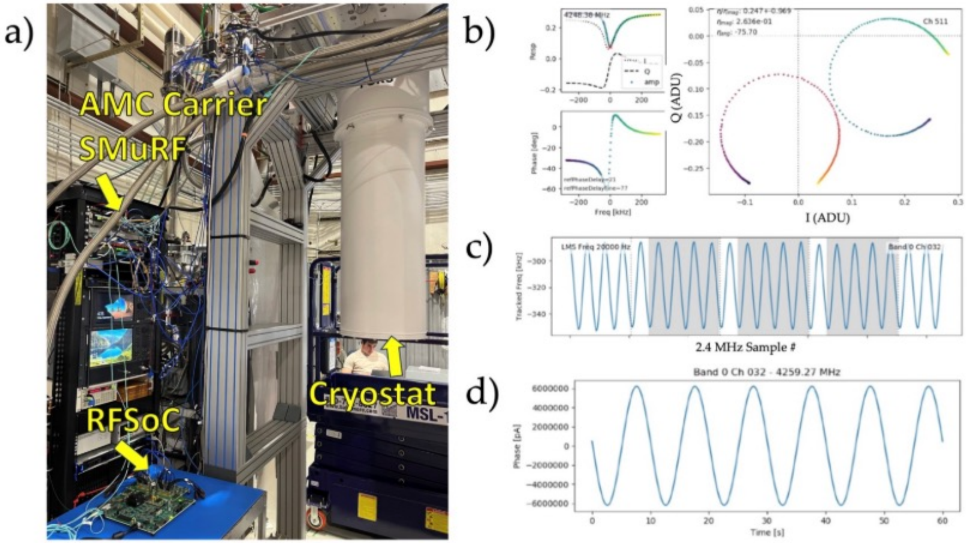


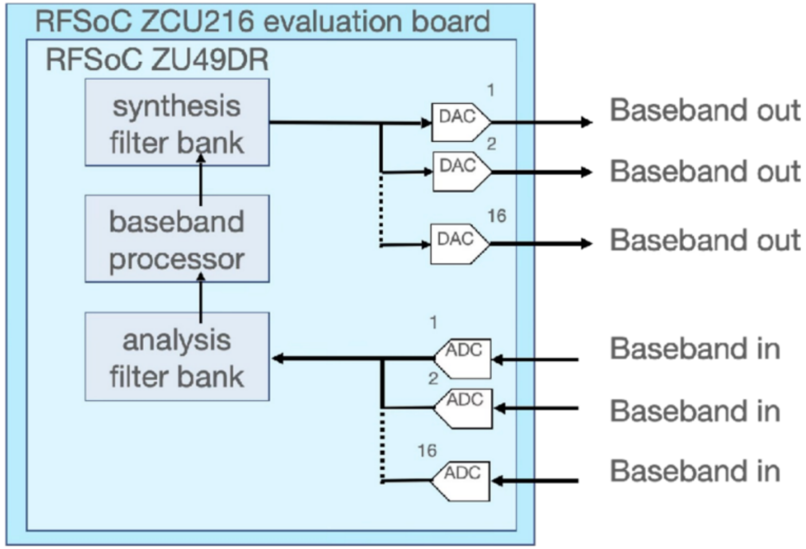
FIG. 2. [Right] Example resonance circle corresponding to an ideal resonance, with magnitude response [Left] for reference. The S_{21}^{min} is denoted with a star in both cases. The color gradient denotes the same points in frequency in both panels. We see that the resonance generically traces out a circle in the complex plane, with the diameter of the circle corresponding to the depth of the resonance.

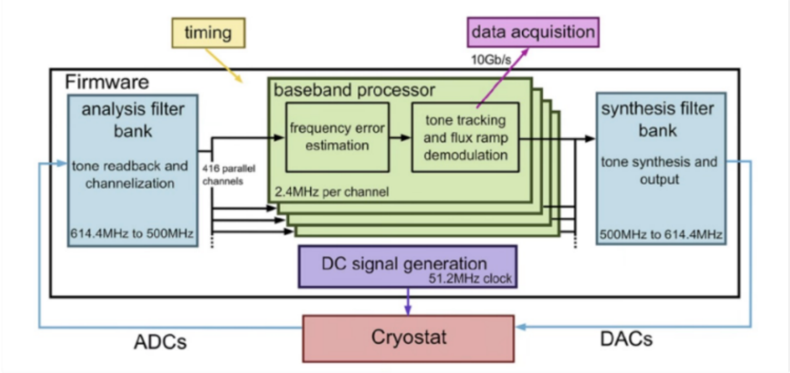




Project 3: SMURF

SMuRF Capabilities



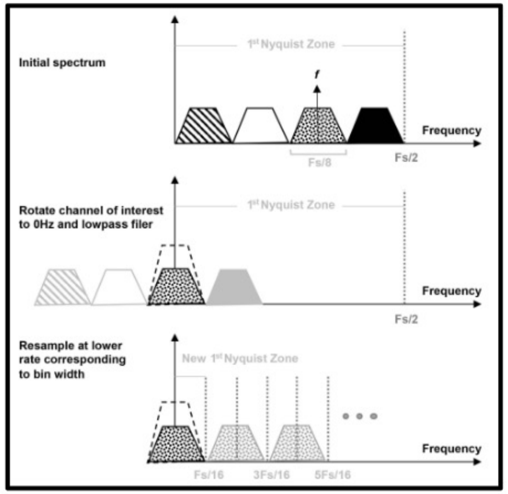


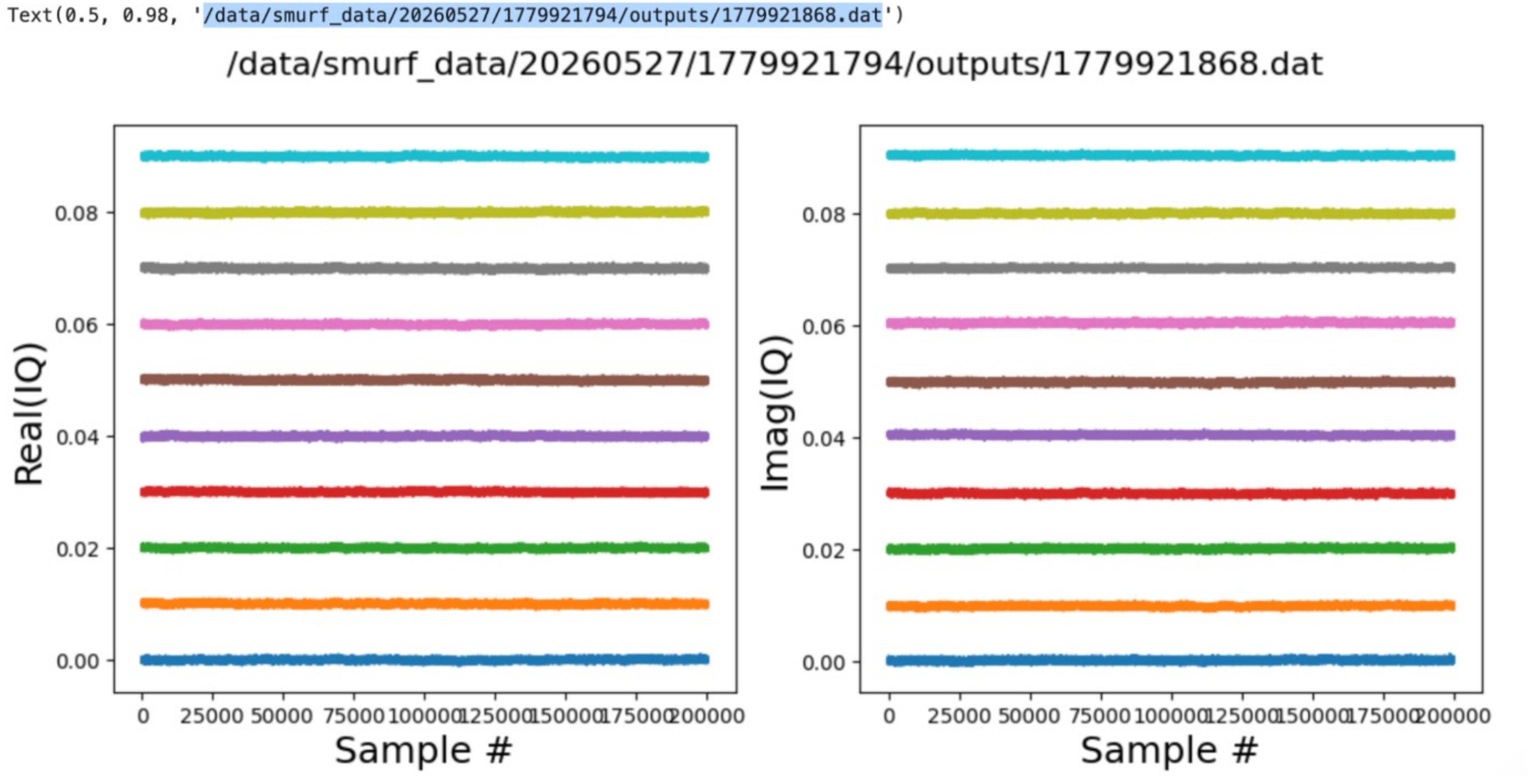
- Tone synthesis and readback
- Tone tracking on FPGA: each tone dynamically follows its resonator, enabling 1000x multiplexing
- External clock synchronization
- Mature software control framework

Project 3: SMURF

IQ Demod & Streaming

- 64 Channels
- 200 kHz

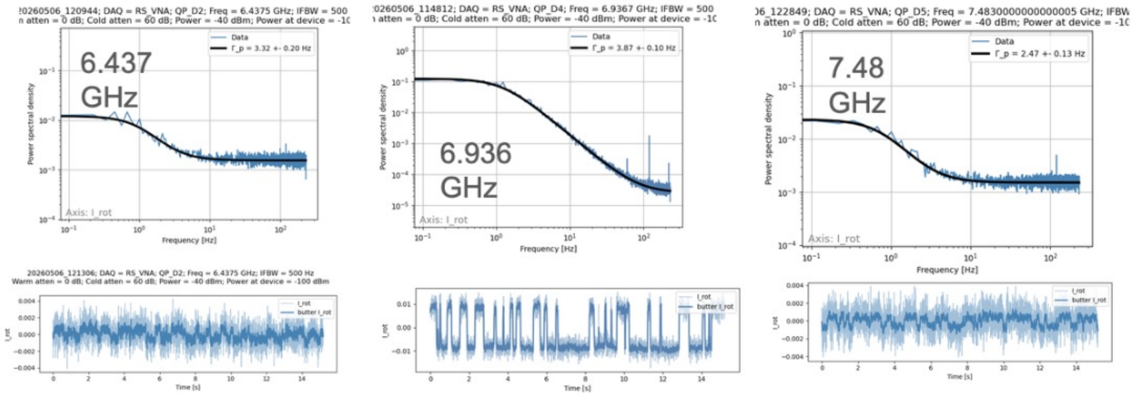


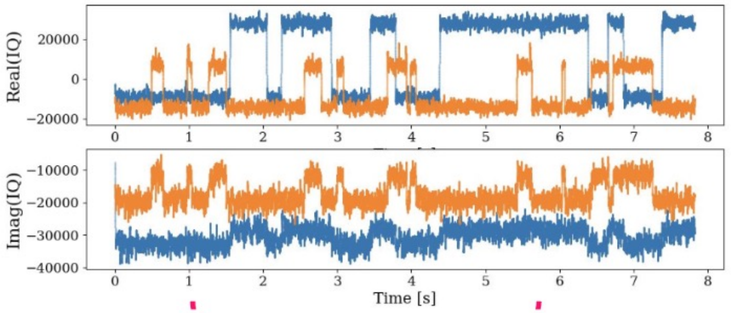


Project 3: SMURF

Multiplexed Readout Development

Have built arrays that have sensors spaced GHz apart → we can build functional chips that can pack ~200 sensors/GHz

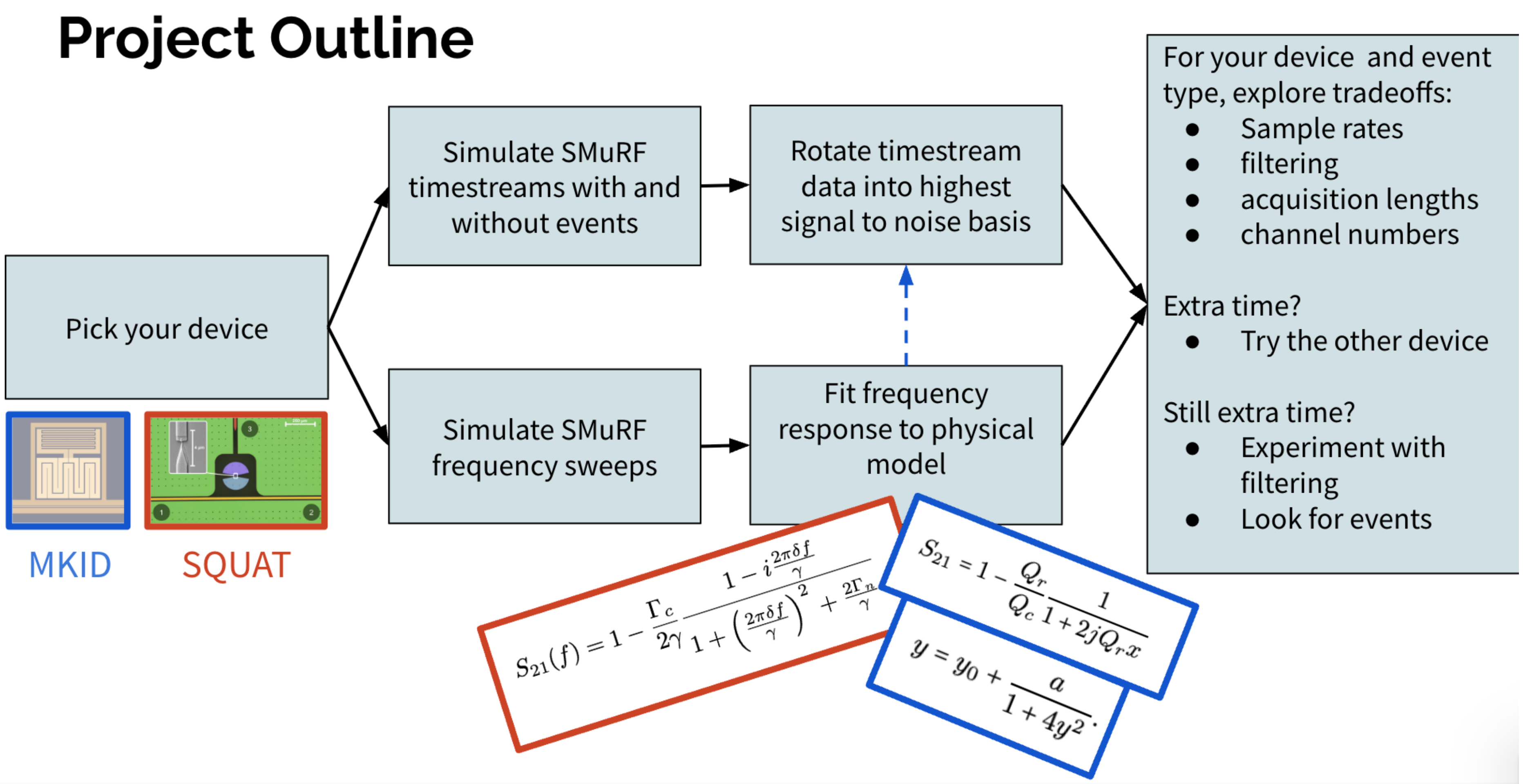




And we can read these simultaneously. Here are two sensors showing some correlated switching events

- Uses Xilinx RFSoC FPGA architecture
- Direct synthesis of 4-8GHz; no analog mixer chain
- Should allow for readout of 100s of qubits simultaneously
- Developing pysmurf codebase for easy interfacing: https://github.com/slaclab/pysmurf/

Project 3: SMURF



Introduction to Josephson Junctions

SLAC Summer Institute

August 2026

Chelsea Bartram, Andrew Yi, Stephen Kuenstner, Pam Stark

The Harmonic Oscillator

- **Motivation:** we can use intuition from a simple harmonic oscillator to understand the behavior of LC circuits
- A sprung mass oscillates by exchanging energy between kinetic and potential energy.
- Adding Josephson junctions to LC circuits makes them anharmonic, tunable, and generally more interesting and useful.

Analogous quantities:

$$\text{(Magnetic Flux)} \quad \hat{\Phi} \to \hat{\mathrm{x}} \quad \text{(position)},$$

$$\text{(Electric Charge)} \quad \hat{\mathrm{Q}} \to \hat{\mathrm{p}} \quad \text{(momentum)},$$

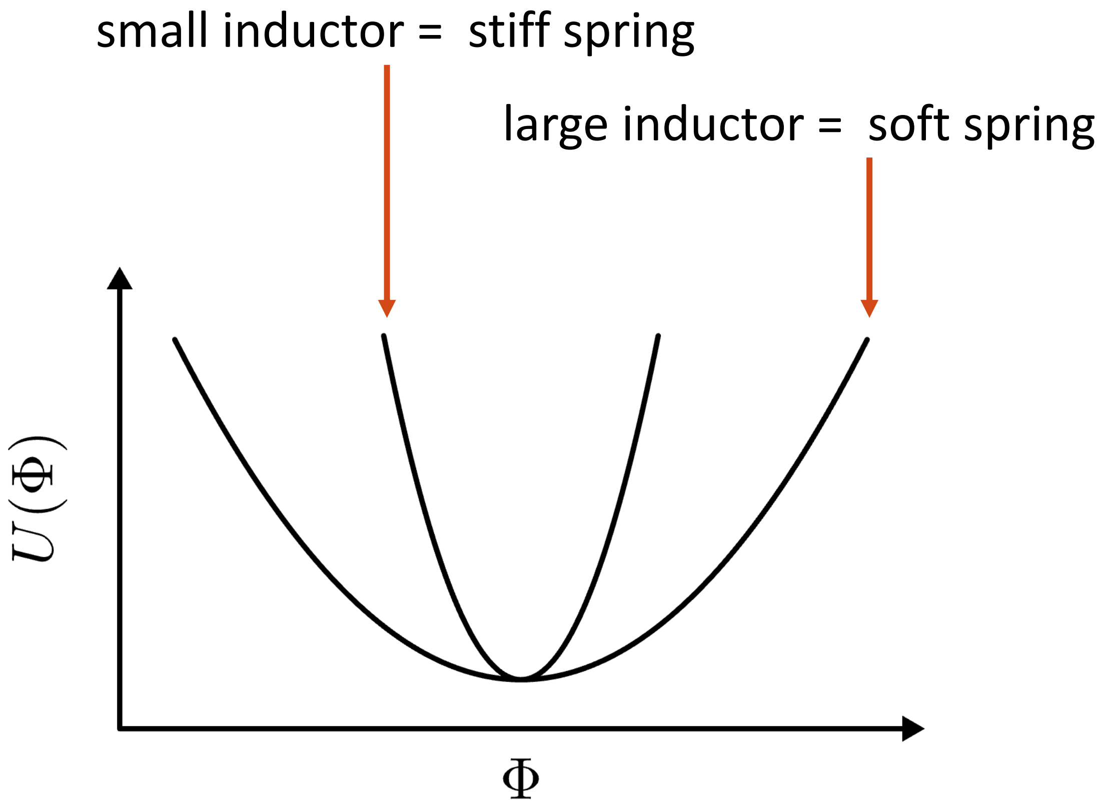
$$\text{(Inductance)} \quad \mathrm{L} \to 1/\mathrm{k} \quad \text{(inverse stiffness)},$$

$$\text{(Capacitance)} \quad \mathrm{C} \to \mathrm{m} \quad \text{(mass)},$$

$$\omega = \sqrt{\frac{1}{\mathrm{LC}}} \to \omega = \sqrt{\frac{\mathrm{k}}{\mathrm{m}}}$$

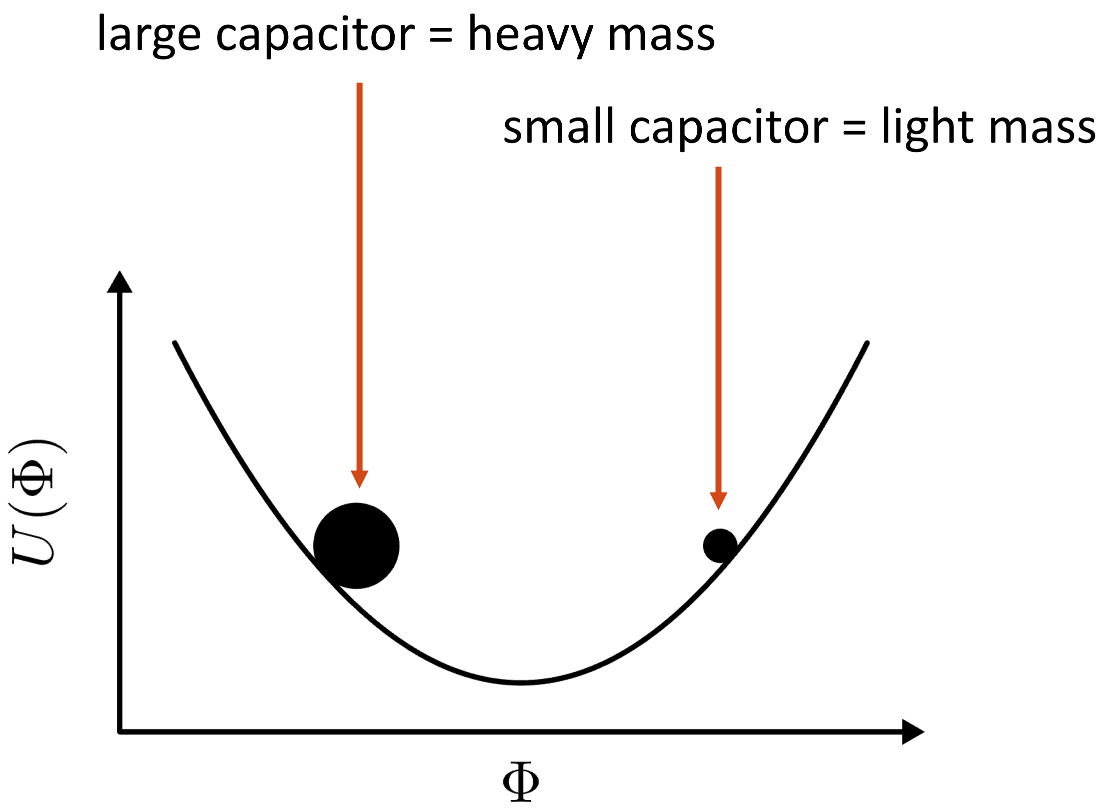
Circuit elements: the inductor

- An inductor stores energy by creating a magnetic field when currents flow through it.
- This will be our potential energy.
- The potential is quadratic in the magnetic flux threading the inductor.
- The inductance sets how much energy is stored per unit flux, just as the spring constant sets how much energy is stored per unit displacement



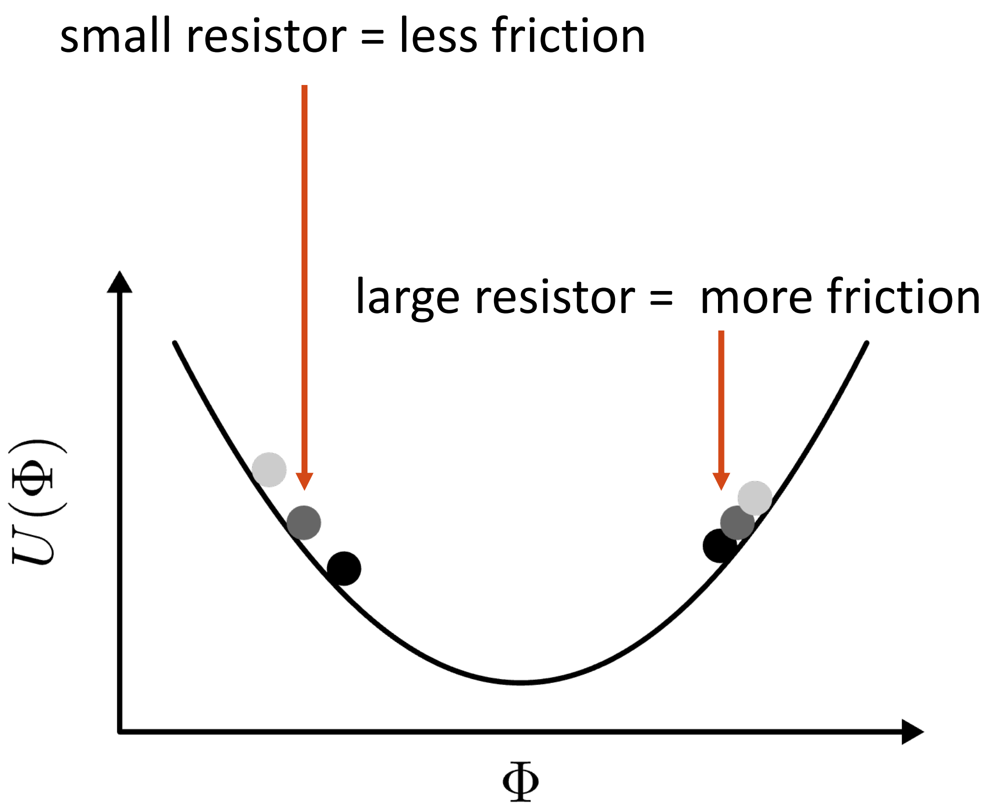
Circuit elements: the capacitor

- A capacitor stores energy by creating an electric field when it is charged.
- In our LC circuit, the charge stored is proportional to the time derivative of the magnetic flux, so it can store kinetic energy.
- The capacitance sets the inertia of the mass.



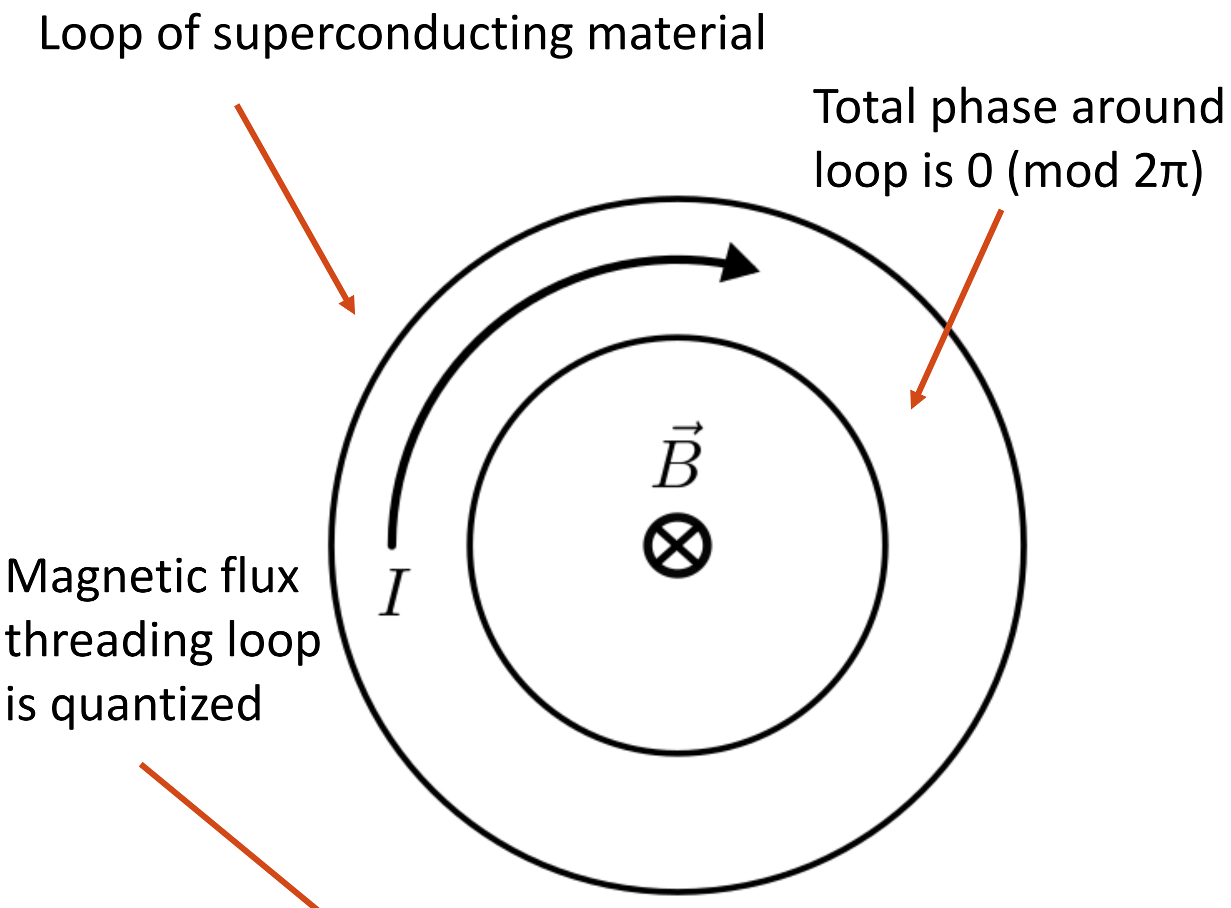
Circuit elements: the resistor

- Resistors and other sources of dissipation play the role of friction, allowing energy to leak out of the circuit.
- Resistors are typically undesirable, as they introduce noise and decoherence.



Superconductivity

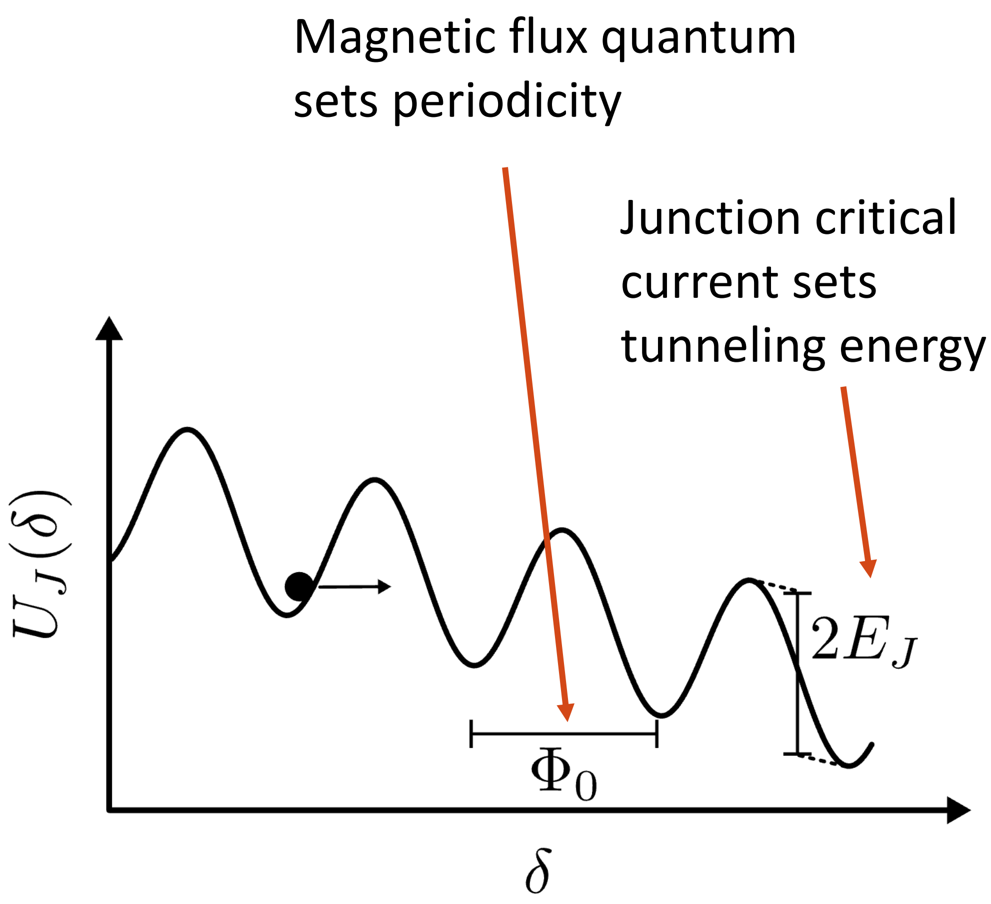
- In a superconductor electrons pair up into Cooper pairs and become bosons.
- The bosons condense into a macroscopically occupied state, which is described by an "order parameter"
- This is essentially a quantum wavefunction, whose magnitude gives the Cooper pair density.
- The order parameter has a phase which must be continuous around any superconducting loop, leading to quantization of magnetic flux!



$$\Phi = 0, \pm\Phi_0, \pm 2\Phi_0, \ldots$$

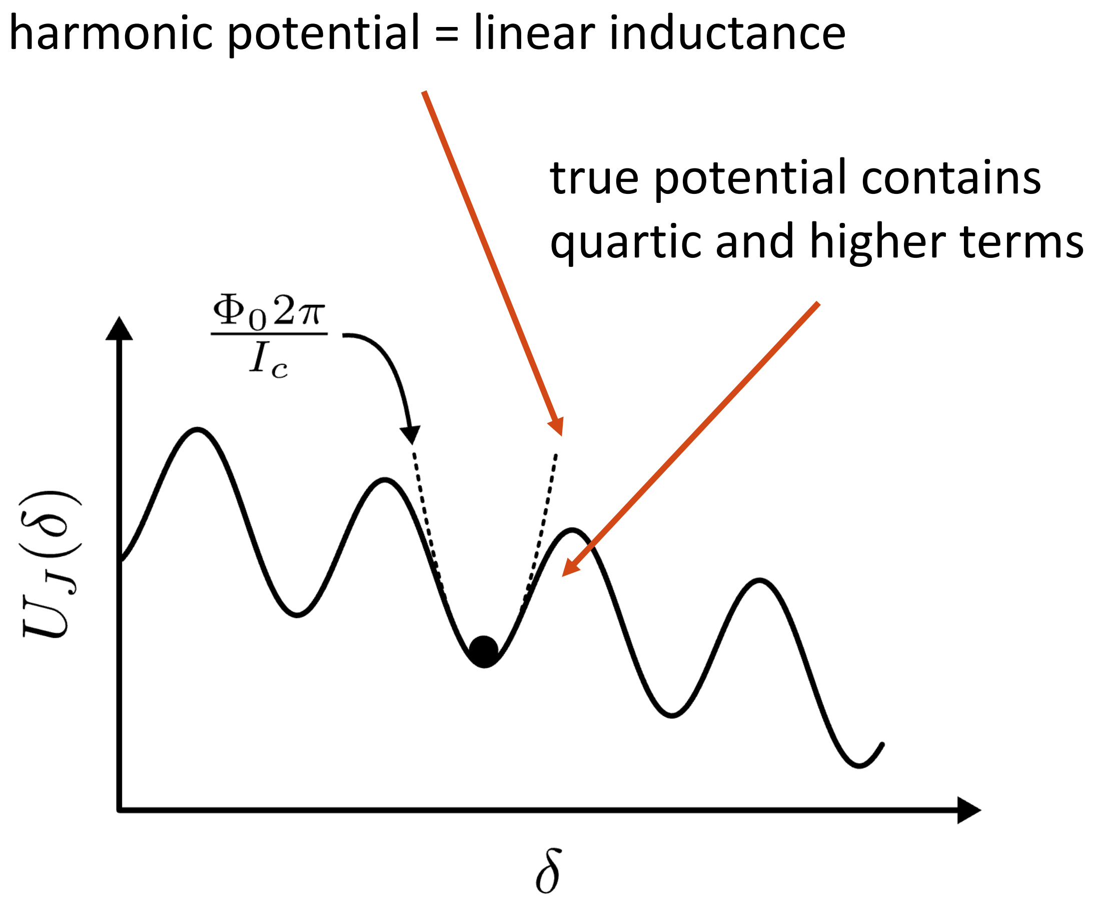
The Josephson junction (finally)

- Just like any quantum particle, Cooper pairs can tunnel through barriers where they would be classically forbidden.
- A superconducting wire interrupted by a barrier is known as a Josephson junction (JJ).
- The potential energy stored in a Josephson junction is a sinusoidal function of the phase difference across the barrier.
- The behavior of a physical junction is well described by the resistively and capacitively shunted junction (RCSJ) model.



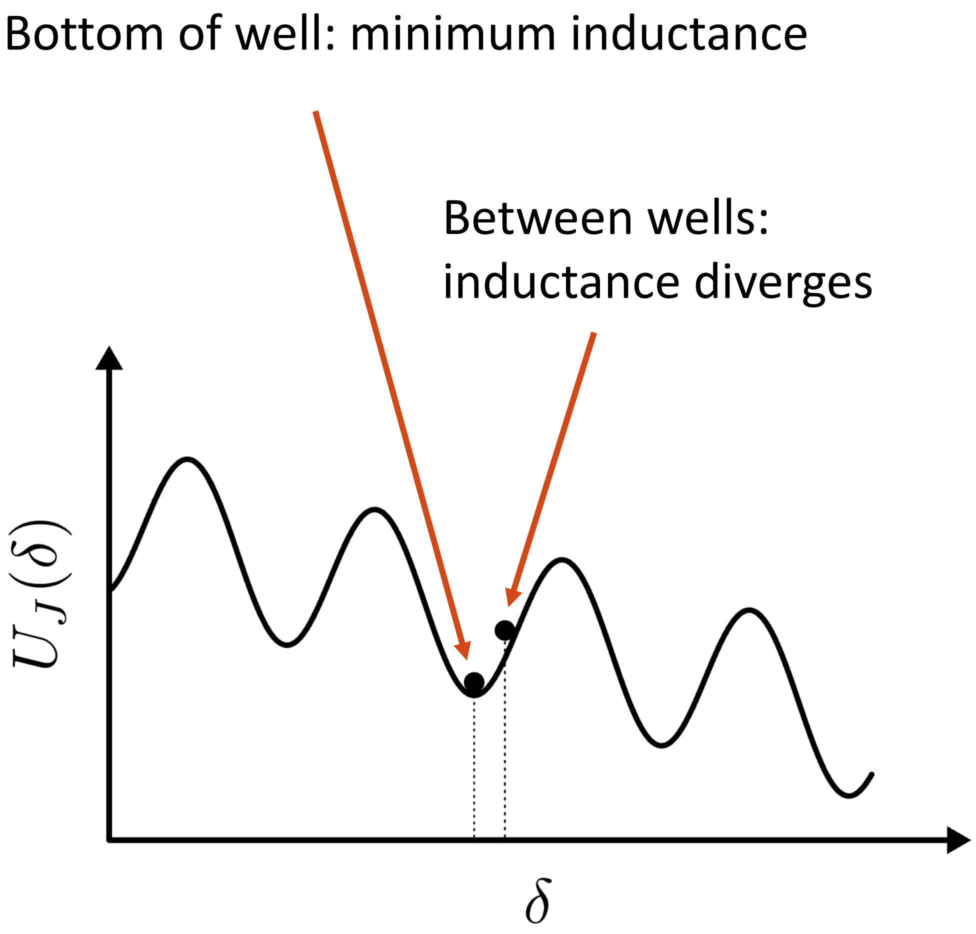
JJ use case #1: anharmonic oscillator

- The potential wells in the washboard potential are strongly anharmonic: in addition to a parabolic term, they contain quartic and higher terms.
- Thus, any circuit containing JJs is nonlinear.
- This is useful because the energy levels of an anharmonic oscillator are not evenly spaced, which allows specific energy levels to be addressed.
- Superconducting qubits use this principle to act as “artificial atoms”.



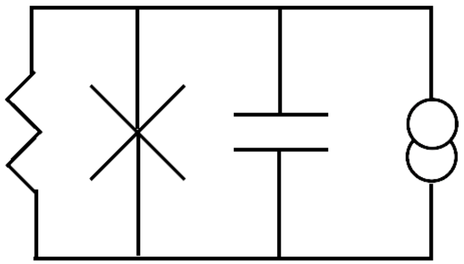
JJ use case #2: tunable inductance

- We can also use a Josephson junction with a tunable phase across them to act as tunable inductances.
- Adding a phase offset changes the curvature, and thus the effective inductance, of the junction.
- This can allow us to create frequency tunable circuits, e.g. rf SQUIDs.



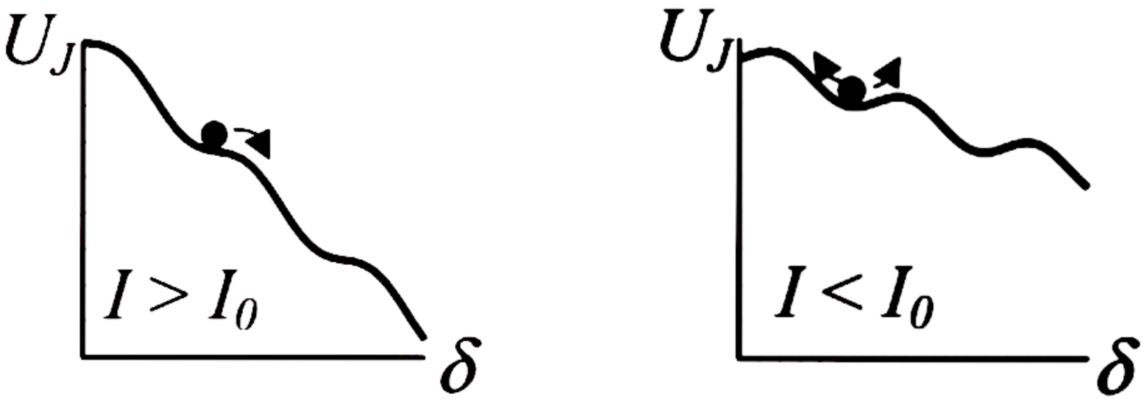
Josephson junction: summary

- Washboard potential: critical current determines tilt



"RSCJ" model

I_0 is the critical current.

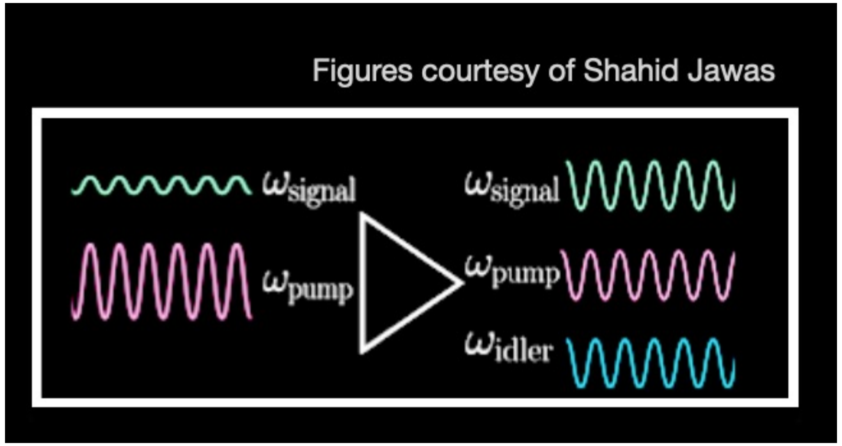


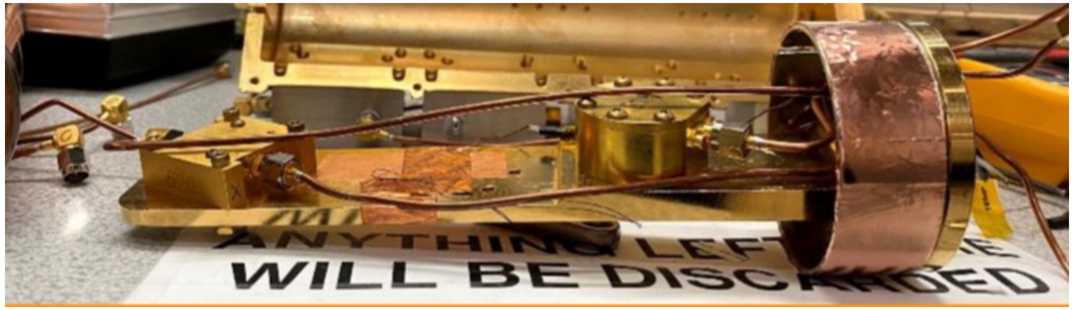
$$\frac{\Phi_0}{2\pi} C\ddot{\delta} + \frac{\Phi_0}{2\pi}\frac{1}{R}\dot{\delta} = I - I_0 \sin\delta = -\frac{2\pi}{\Phi_0}\frac{\partial U_J}{\partial \delta}$$

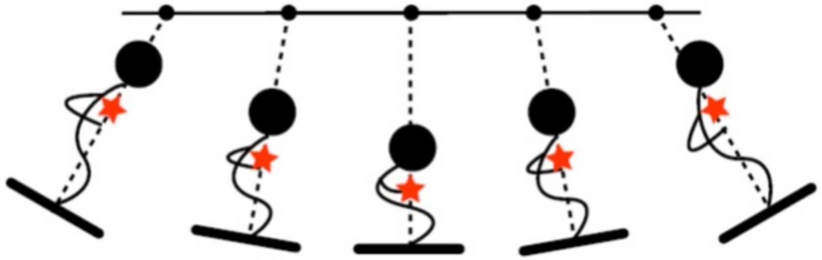
$$U_J \equiv \frac{\Phi_0}{2\pi}\{I_0(1 - \cos\delta) - I\delta\} = E_J\{1 - \cos\delta - i\delta\}$$

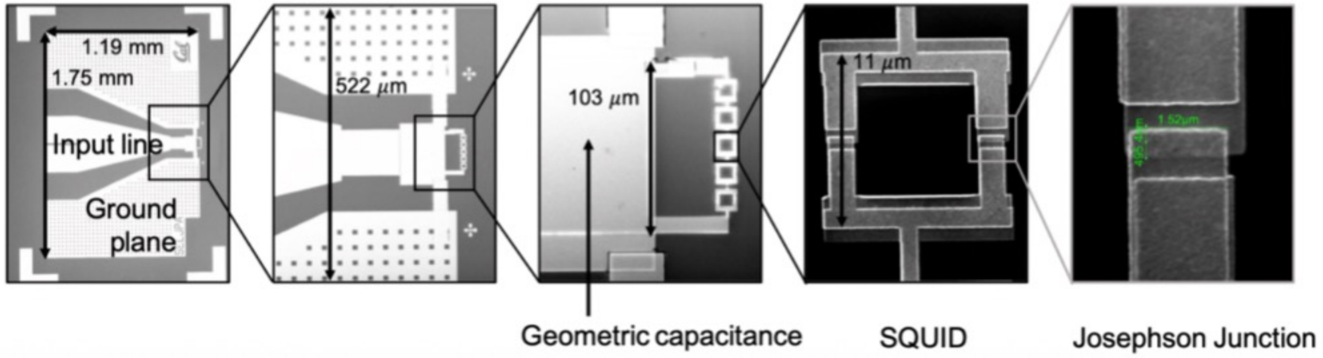
Continuous wave amplification

Josephson parametric amplifiers

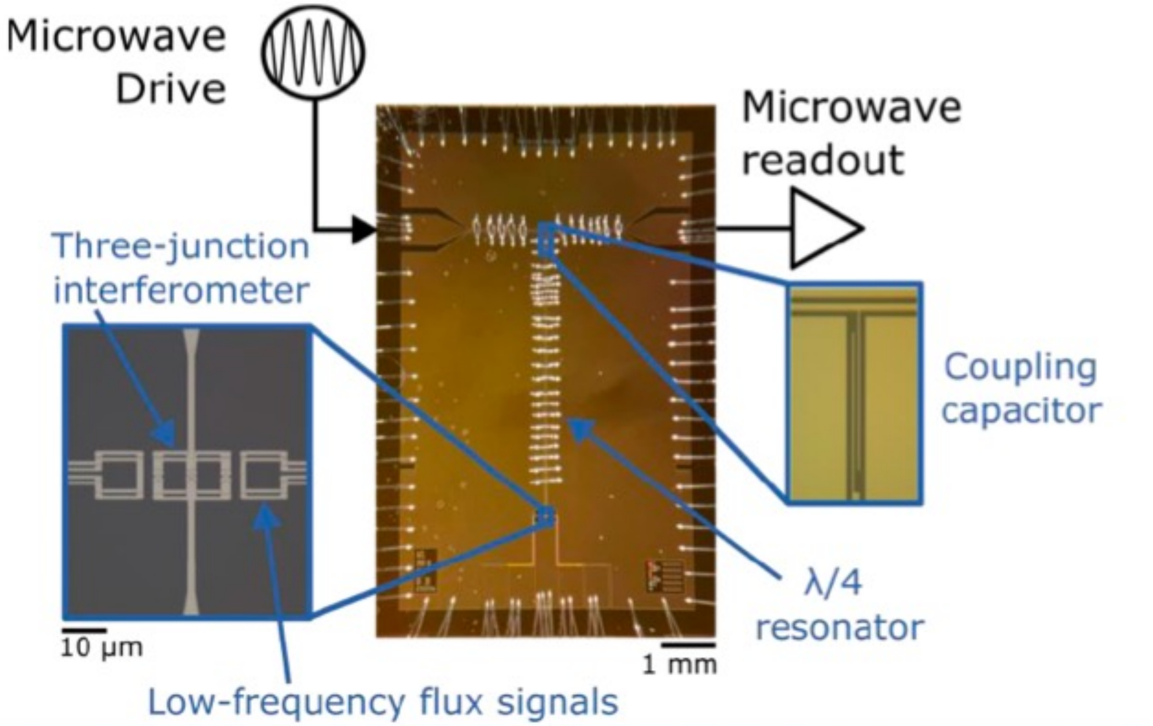


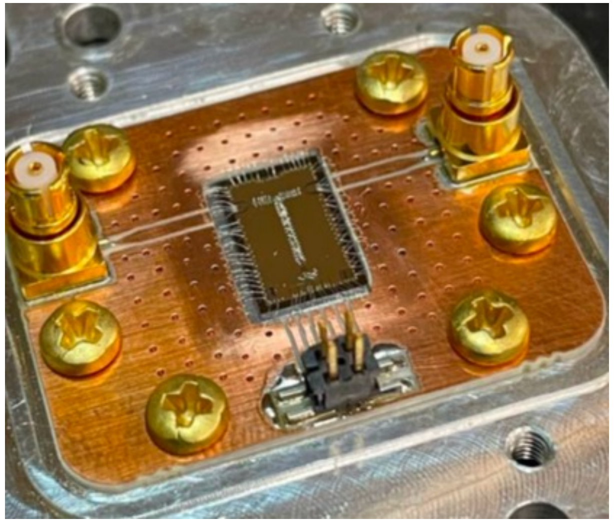






SQUIDs and RF Quantum Upconverters





RF Upconverters courtesy of Kent Irwin's group

Suggested reading

- Clarke, John, and Alex I. Braginski. *The SQUID handbook: fundamentals and technology of SQUIDs and SQUID systems*. John Wiley & Sons, 2006.
- Girvin, Steven M. "Circuit QED: superconducting qubits coupled to microwave photons." *Quantum machines: measurement and control of engineered quantum systems* 96 (2014).
- Blais, Alexandre, et al. "Quantum-information processing with circuit quantum electrodynamics." *Physical Review A—Atomic, Molecular, and Optical Physics* 75.3 (2007).

Non-linearity of the Josephson junction

- The non-linear inductance of the Josephson junction is a very important feature!
- Most Josephson-junction-based amplifiers leverage this non-linearity for parametric amplification

Deriving the inductance of the Josephson junction.

$$v(t) = L\frac{di}{dt}$$

If we can write down the current in the junction, we can derive the inductance.

$$I = I_0 \sin\phi$$

Know junction current as a function of phase.

$$\frac{di}{dt} = I_0 \cos(\phi)\dot{\phi}$$

← This looks like the definition of voltage

$$V = d\phi/dt$$

Identify the rest as 1/L

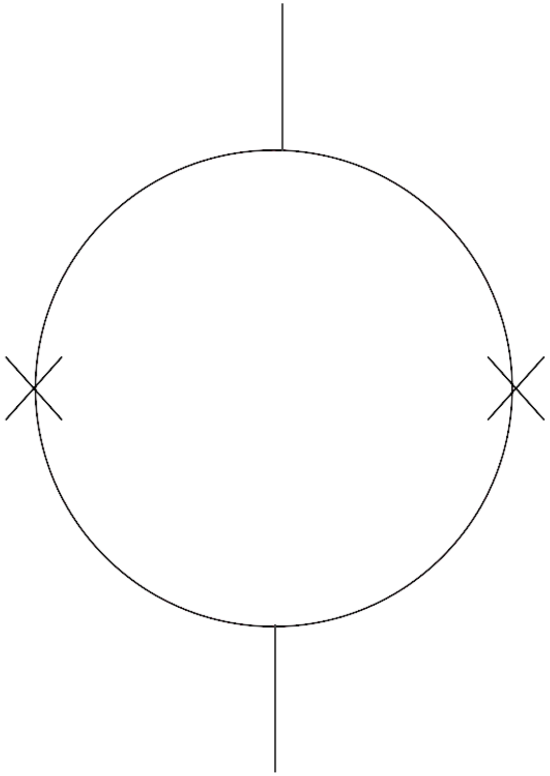
$$L \propto \frac{\Phi_0}{2\pi I_c \cos\phi}$$

Derive the inductance of a single junction

- We know that the $V = -L\, dI/dt$.
- Use the constitutive equations to derive L, the inductance, for a single junction of critical current I_c.
- Use the same equations to derive L, the inductance, for two junctions in parallel with critical currents I_{c1} and I_{c2}, and phases ϕ_1 and ϕ_2 respectively. What happens when $I_{c1}=I_{c2}$?
- Use the same equations to derive L, the inductance, for two junctions in series, with critical currents I_{c1} and I_{c2}, and phases ϕ_1 and ϕ_2 respectively. What happens when $I_{c1}=I_{c2}$?

Building a SQUID

- Two Josephson junctions connected in parallel
- Very sensitive interferometer capable of measuring signals ~5x10^{-18} T
- In the absence of a magnetic field, current splits equally
- In presence of a field, a screening current circulates with an associated Josephon phase that is associated with the magnetic field
- As soon as the current exceeds the critical current on either side, a voltage appears across a junction



Project #1: Simulating Josephson junctions with WRSpice

- SPICE is a well-known circuit simulation tool. The original SPICE allows modeling of standard circuit components.
- Standard inputs: circuit design file
- Standard outputs: currents and voltages throughout the circuit
- WRSPICE is open source code for modeling circuits with Josephson junctions. It includes modeling capabilities for all the standard SPICE components plus the ability to model Josephson junctions.
- The biggest difference between SPICE and WRSPICE is it has to track this new circuit parameter, which is the phase of the waveform.

Project #1: Getting started

- First, navigate to website: http://wrcad.com/wrspice.html. Read the short description of the software on that main page.
- This site has the documentation for wrspice, including a comprehensive manual. Click on the link to the manual and you will find a long table of contents page.
- In the table of contents, navigate to http://wrcad.com/manual/wrsmanual/node142.html
- This is the page on Josephson junctions. Take some time to peruse this. The "Josephson junction description" is particularly interesting. Take some notes; write some "fun facts" down to share with the larger group.

Project #1: Open the terminal

- Now go to your desktop and navigate to the search bar.
- Type “Msys”. You should see several options pop up.
- Select the one that says Msys2 Mingw64. It should be a blue color.
- This terminal environment emulates a linux environment. You can now issue standard linux commands.
- Navigate to the wrspice example folder with

  cd /usr/local/xictools/wrspice/examples

- Execute an example using the wrspice command followed by the circuit file

  wrspice jjac.cir

Project #1: Some examples to test

- Now go to your desktop and navigate to the search bar.
- Type “Msys”. You should see several options pop up.
- Select the one that says Msys2 Mingw64. It should be a blue color.
- This terminal environment emulates a linux environment. You can now issue standard linux commands.
- Navigate to the wrspice example folder with

  cd /usr/local/xictools/wrspice/examples/JJexamples
- Execute an example using the wrspice command followed by the circuit file

  wrspice jjac.cir

Project #1: Study the examples

- Try running the following examples and take some notes on the output:
  - jjIV.cir: This generates an IV curve for a Josephson junction. What do you notice?
  - jjac.cir:
  - jjdc.cir:

- If there is time, explore the other circuit files and write down some notes on your favorite.

Project #1: Now let’s build something more challenging

- Can we build a SQUID out of Josephson junctions in WRSPICE?
- In the terminal, navigate to your home directory
- Navigate to the directory SSI_QIS
- Type ‘jupyter notebook’ to open a notebook in the browser tab.
- In the browser tab, you should see a notebook called ‘squid_demo’. Click on it.
- The code writes a circuit file to represent a SQUID. Look at the circuit file and translate it into a drawing of a circuit diagram, with labels.
- Run the notebook. The final plot should take about 5 minutes to generate. It should save the last plot as a PDF.

Project #1: What is the code doing?

- The code is looping over circuit file parameters. Write down which circuit file parameters are being changed each time wrspice is called.
- What type of WRSPICE analysis is it doing?
- What does the final plot show?
- Trying changing:
  - the values of the different circuit parameters
  - the range over which you loop

Note the effects of these changes; save the plots as separate files and/or screenshots and write down what you observe.

Project #1: We just learned of a WRSPICE alternative!

- Challenges of WRSPICE: Complex circuits with large numbers of Josephson junctions means many more parameters to track. WRSPICE has some parallel capabilities but not a lot.
- Alternative option courtesy of Kevin O'Brien, that can be 10,000x as fast: https://github.com/kpobrien/JosephsonCircuits.jl
- Try a git clone in the home directory?

G4CMP Tutorial

For tomorrow, since there is less time to get started…

G4CMP Tutorial

1. Getting started
   - How to build an application, visualize and plot data

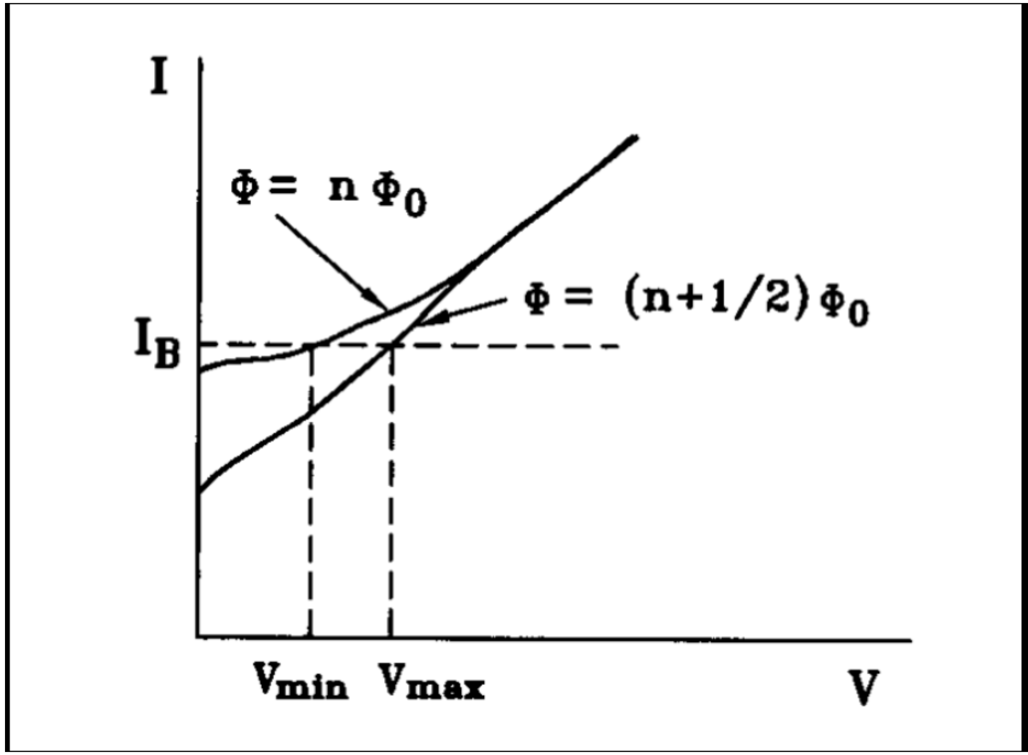
2. Making modifications
   - Changing sensor properties and device geometries

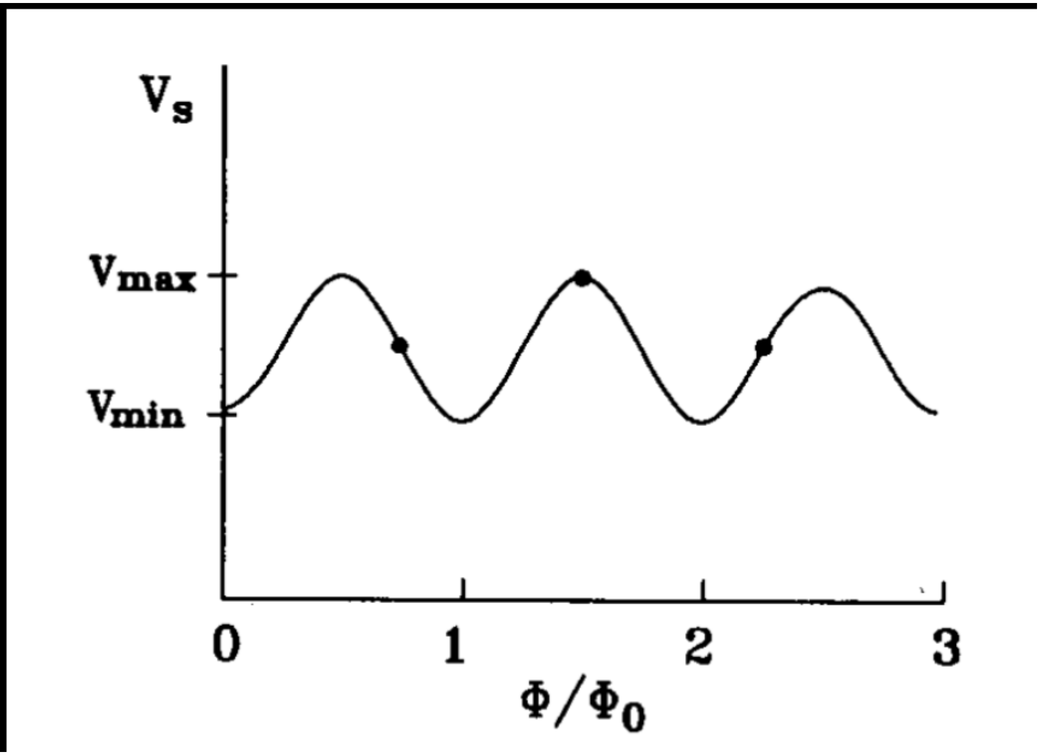
3. Adding more physics
   - Adding custom physics for interactions with primaries such as gammas, neutrons, etc.

G4CMP Tutorial

How to use your own computer to run the tutorial:

1. Install Docker desktop
2. $Docker pull cstonewhitehead/slac-summer-institute:tutorial
3. Install vcxsvr (X server for visualization)
   - VcXsrv Windows X Server download | SourceForge.net
4. Start the X server with XLaunch application
5. $Docker run –it –p 8888:8888 cstonewhitehead/slac-summer-institute:tutorial


I
$\Phi = n\,\Phi_0$
$\Phi = (n+1/2)\,\Phi_0$
I_B
V_{min} V_{max}
V


V_s
V_{max}
V_{min}
0
1
2
3
Φ/Φ_0