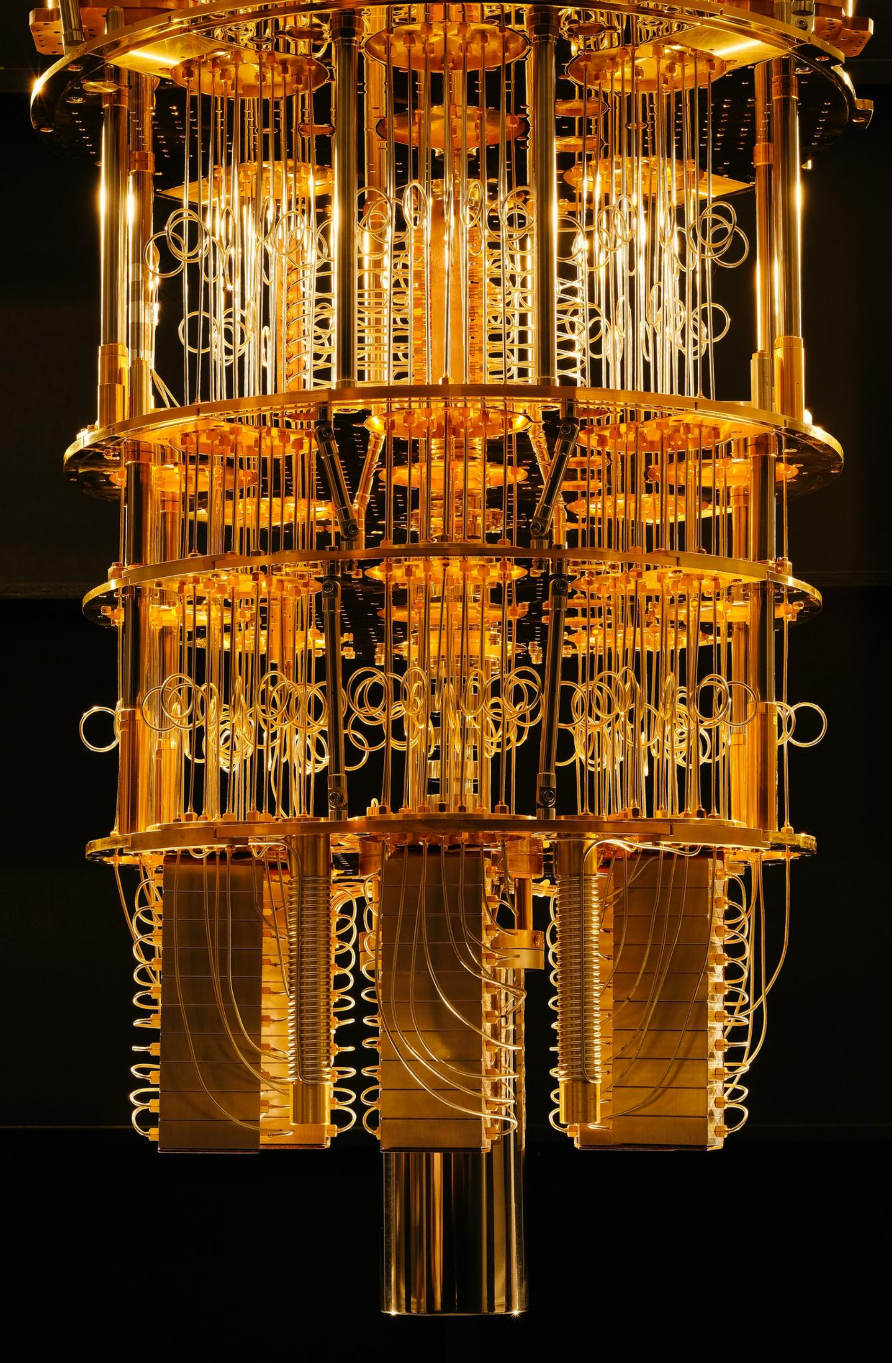


FPGA Qubit Package

6.111 Final Project Proposal

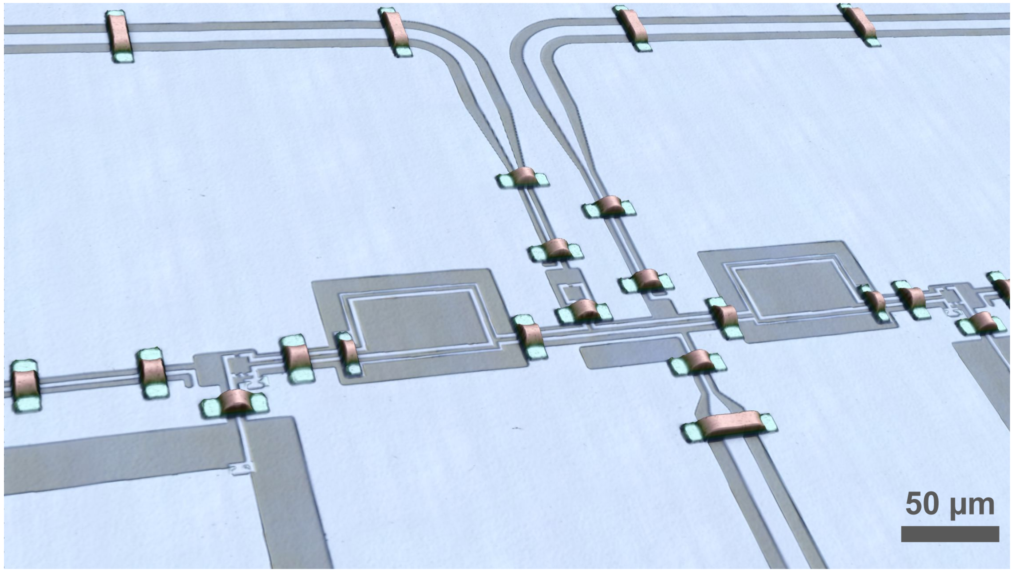
Francisca Vasconcelos and Megan Yamoah

Engineering Quantum Systems (EQuS) Group
Massachusetts Institute of Technology

6 November 2018

Introduction

- Implement useful pre-processing for superconducting quantum devices
- Involves:
  - Processing microwave signals from devices at ~20 mK
  - Classifying state
  - Outputting data in a meaningful format
- Why FPGA?
  - Feedback
  - Quantum algorithms



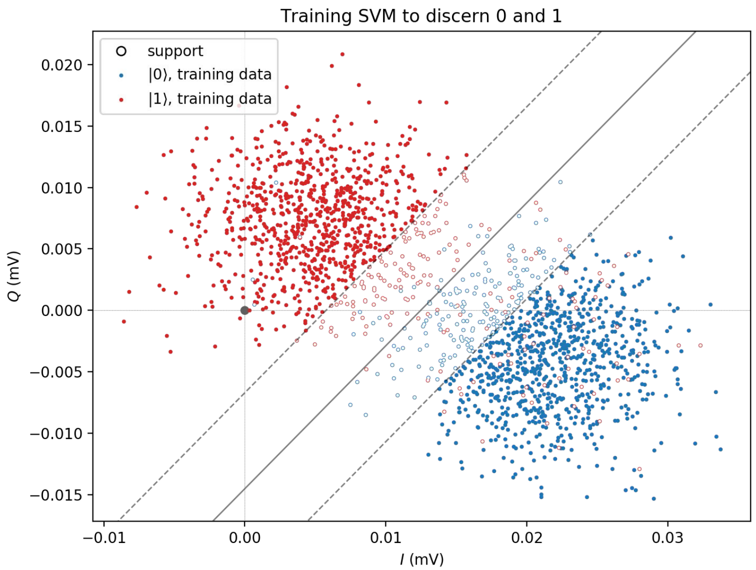
J. Yoder, D. Kim, et al., (2016)

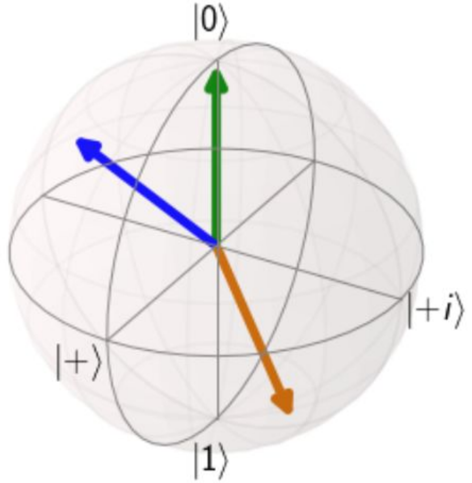
Background

- **Qubit state represented in 2D Hilbert space**
  - **Complex number space**
  - **Superpositions**
- **Measurement**
  - **Single basis**
  - **Wave function collapse**
  - **Statistical**
- **State discrimination**

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$

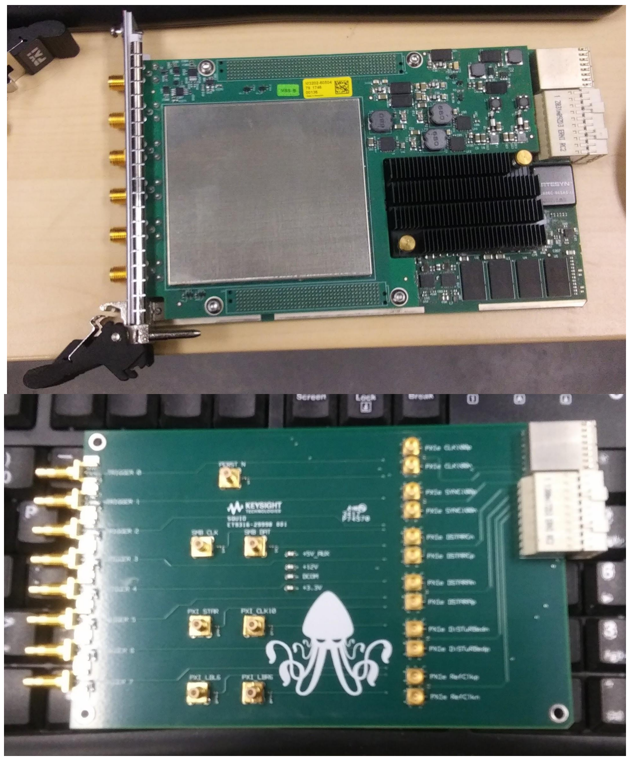
$$\alpha, \beta \in \mathbb{C}$$





Lab Equipment

Xilinx Kintex-7 with Keysight compatibility



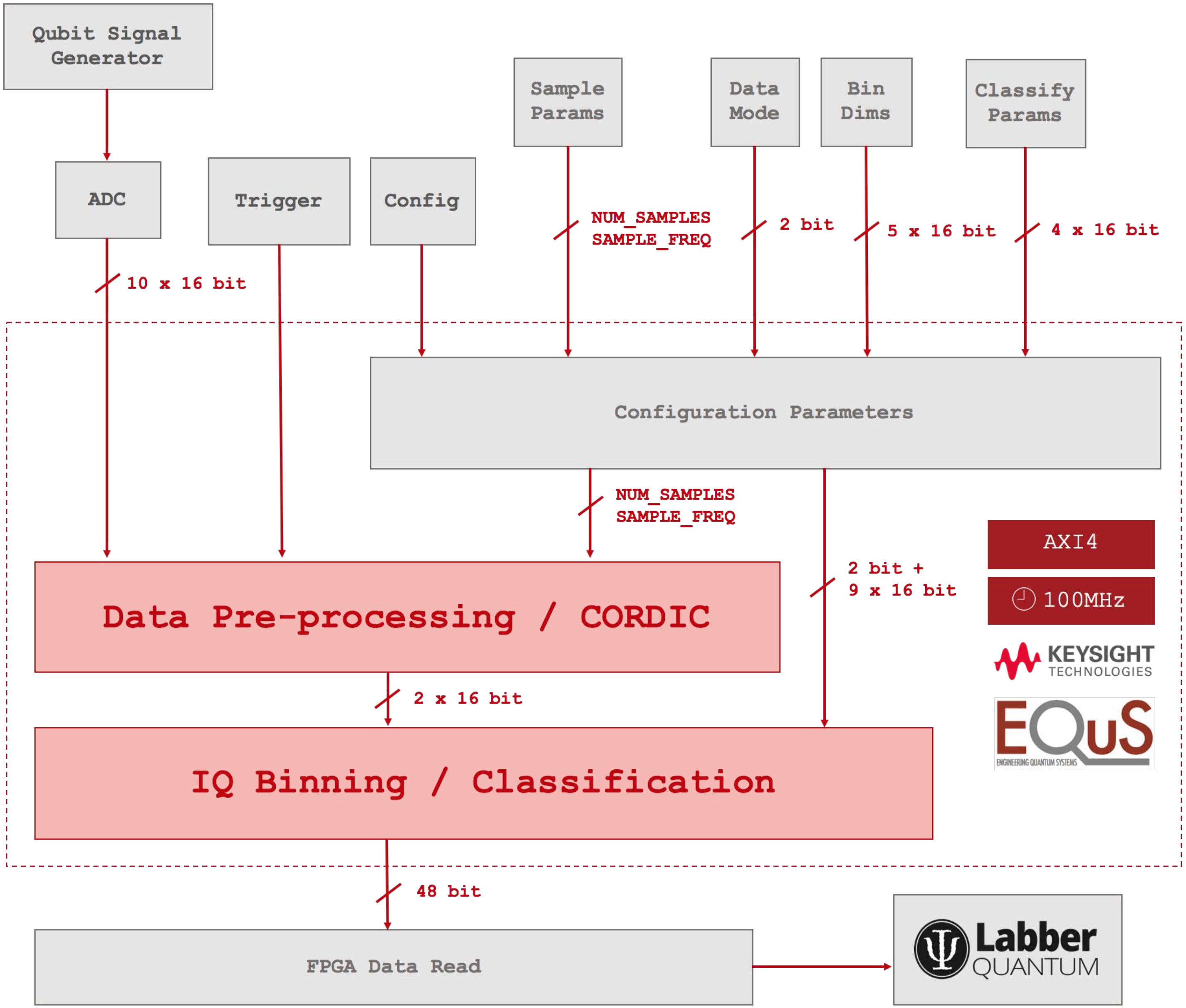
AWG

Keysight PXI Enclosure

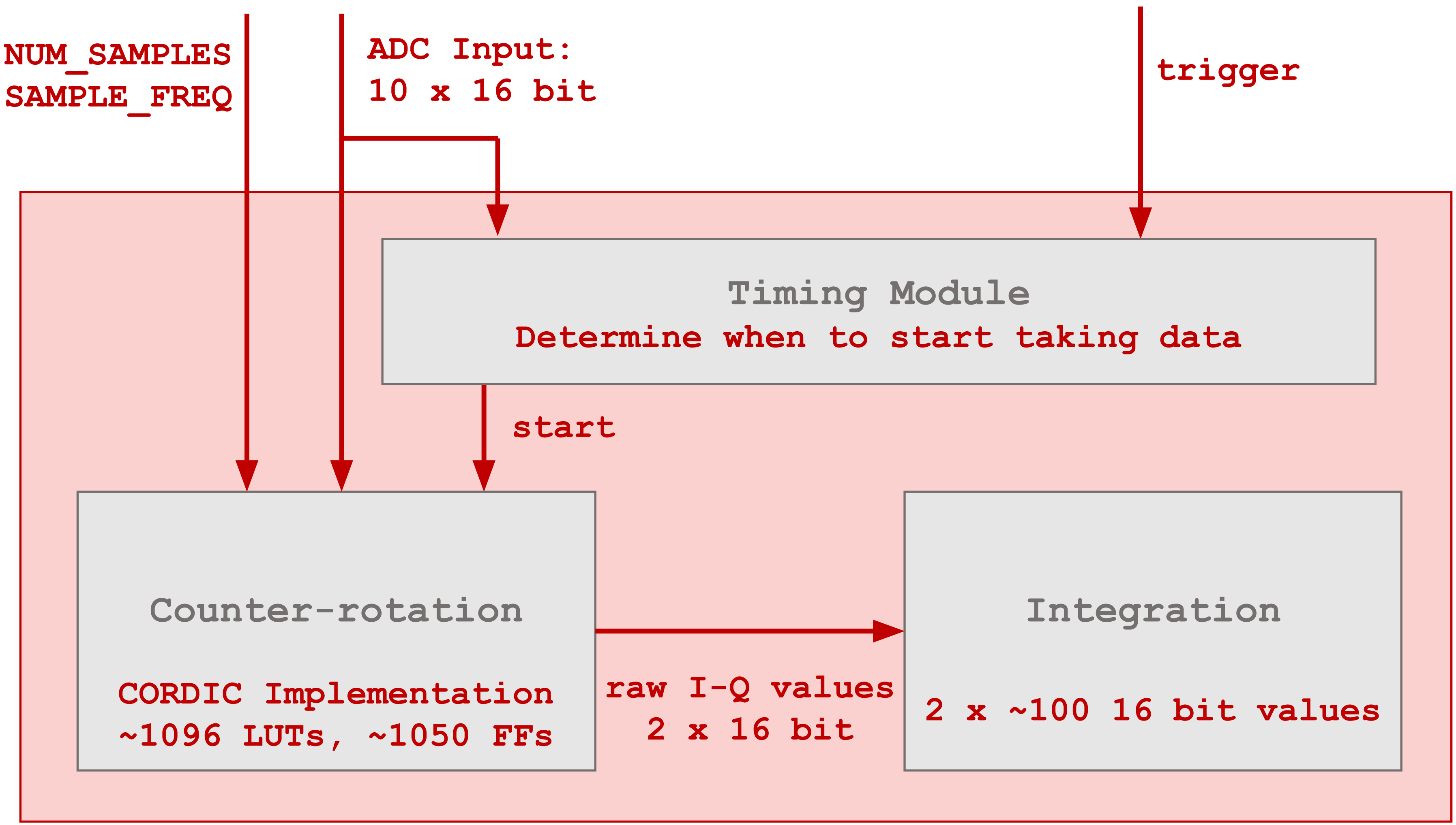
Cryogenic Fridges with Qubit Packages

Lab Measurement Racks

Broad Block Diagram



Pre-processing Diagram



Initial Processing - Background

Qubit output

$$\boxed{s(t) = cos(\omega_{RF}t + \theta)}$$

I-Q mixer output

$$\begin{cases} \mathrm{I(t)} = \cos(\omega_{IF}t + \theta) \\ \mathrm{Q(t)} = \sin(\omega_{IF}t + \theta) \end{cases}$$

Complex number representation

$$\mathrm{z(t)} = \mathrm{I(t)} + \mathrm{iQ(t)}$$

$$z(t)e^{-i\omega_{IF}t} = I(t)\cos\omega_{IF}t + Q(t)\sin\omega_{IF}t + i\left(Q(t)\cos\omega_{IF}t - I(t)\sin\omega_{IF}t\right)$$

$$\boxed{\begin{aligned} \mathrm{I} &= \mathrm{I(t)}\cos\omega_{IF}t + Q(t)\sin\omega_{IF}t = \cos\theta \\ \mathrm{Q} &= \mathrm{Q(t)}\cos\omega_{IF}t - I(t)\sin\omega_{IF}t = \sin\theta \end{aligned}}$$

COordinate Rotation DIgital Computer

- **Iterative algorithm**
- **Represent rotation matrix as multiplication by a power of 2 or adding operations**

Vector rotation →

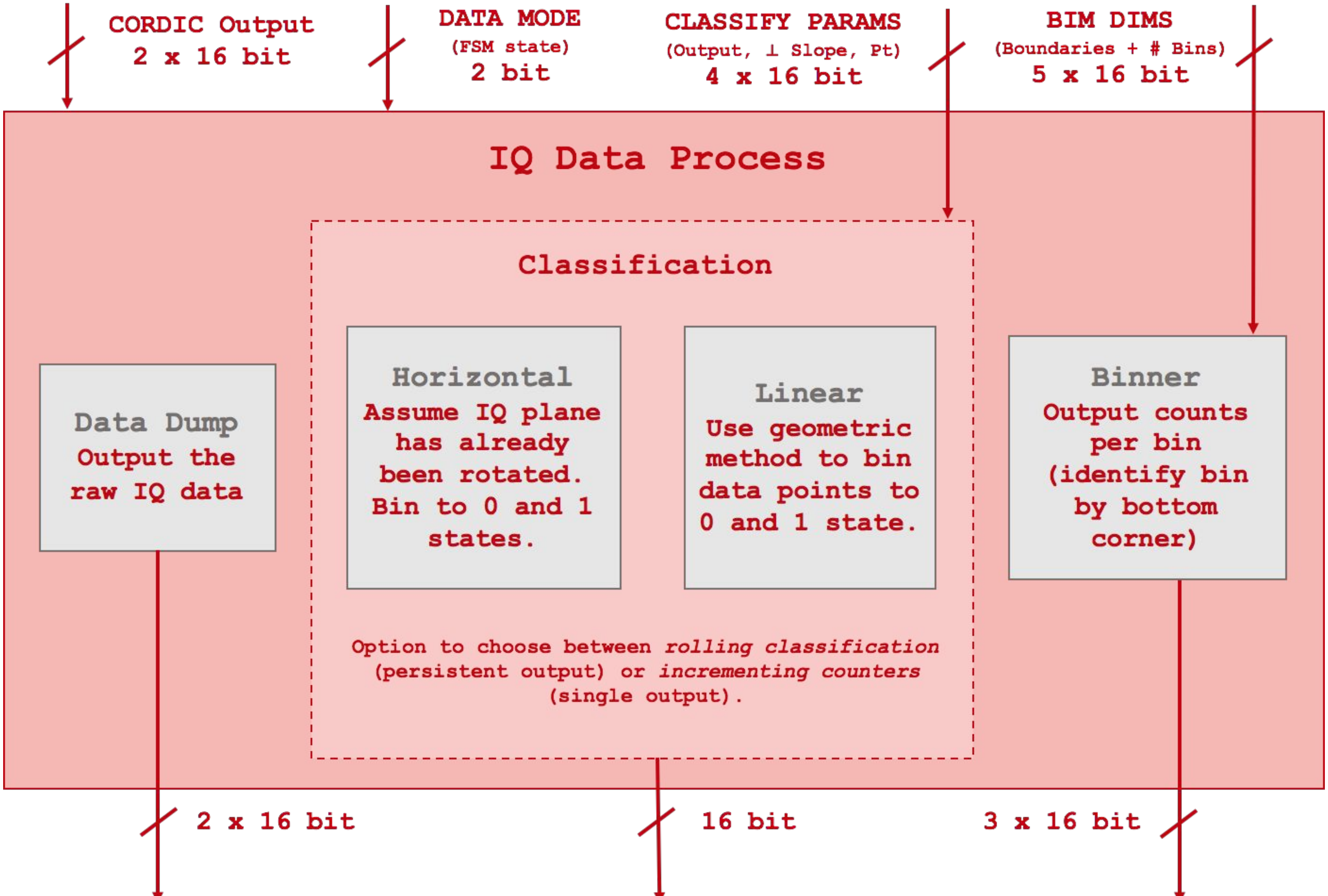
$$\boxed{\begin{aligned} x' &= x\cos\theta - y\sin\theta \\ y' &= y\cos\theta + x\sin\theta \end{aligned}}$$

CORDIC

$$\left\{\begin{aligned} x' &= \prod_{i=1}^{n}\cos(\arctan(2^{-i}))(x_i - \alpha_i y_i 2^{-i}) \\ y' &= \prod_{i=1}^{n}\cos(\arctan(2^{-i}))(y_i + \alpha_i x_i 2^{-i}) \\ \theta' &= \sum_{i=1}^{n}\theta - (\alpha_i \arctan(2^{-i})) \end{aligned}\right.$$

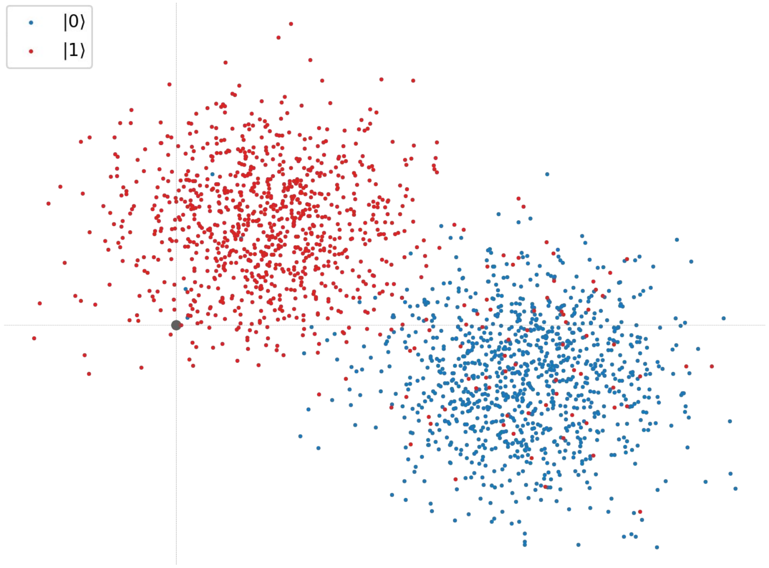
https://www.xilinx.com/support/documentation/ip_documentation/cordic/v6_0/pg105-cordic.pdf

Binning / Classification Diagram

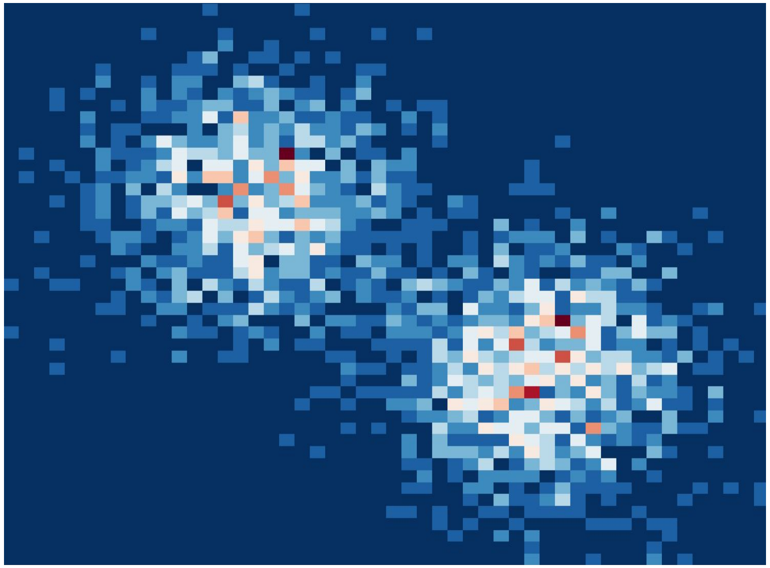


Binner

- Divide I-Q plane into squares (user defines resolution)
- Continuously bin values into this 2D histogram
- Once measurement is over, report count for each bin
- Decreased resolution reduces amount of data sent to computer & increases processing speed



Actual Data

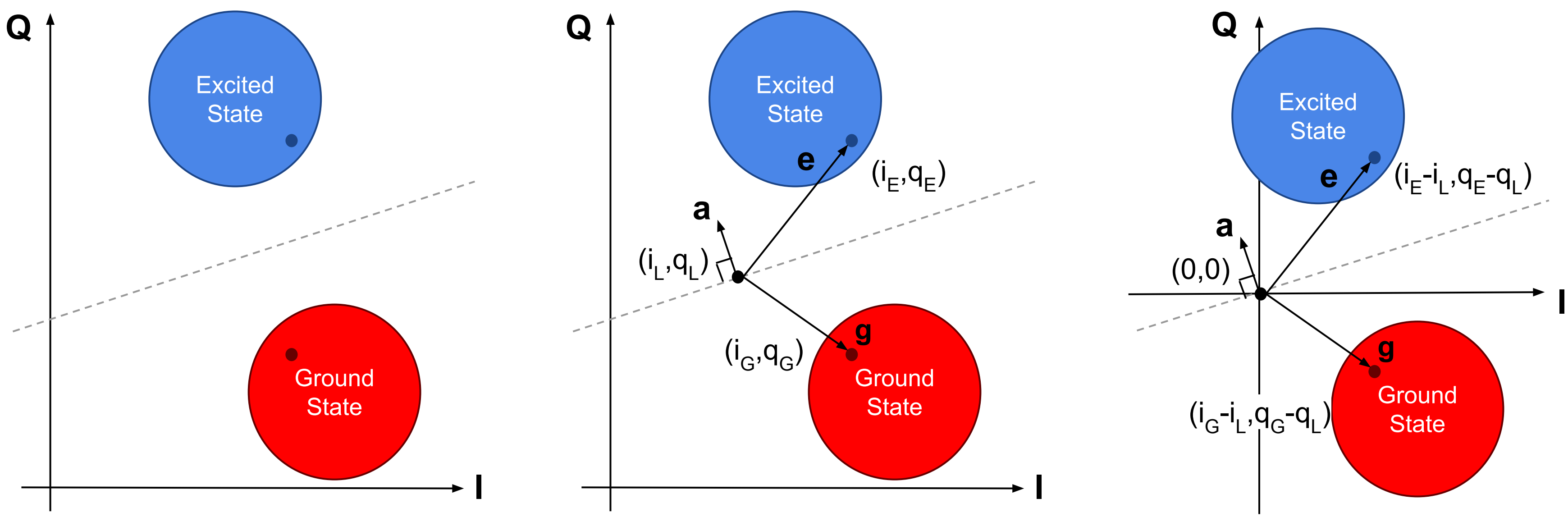


Binned Data

(Specifiable # Bins & Bin Width)

Classification

- **Easier: Horizontal (assume I-Q mapping has been rotated)**
- **Harder: Linear with Proposed approach: Geometric Method**
  (No need to use arctan, sqrt, or any other complicated functions!)



1) Get values to be classified

2) Create vector representation

3) Map vectors to origin

4) Compute dot products and accordingly bin values:

$$\vec{a}\cdot\vec{b}:\begin{cases} >0, & if\ \vec{b}=\vec{e}\ \rightarrow\ \boldsymbol{bin\ as\ 1} \\ =0, & if\ point\ on\ classifier \\ <0, & if\ \vec{b}=\vec{g}\ \rightarrow\ \boldsymbol{bin\ as\ 0} \end{cases}$$

Goals & Timeline

Megan's Goals

- Baseline
  - Receive and output digitized signal values
  - Output random I-Q values
- Expected
  - CORDIC rotation
  - Integration / Averaging
- Stretch
  - De-noising
  - Integrate in lab measurement chain

Fran's Goals

- Baseline
  - Data dump
  - Data read
- Expected
  - Horizontal classifier
  - Configuration parameters
  - Binner
  - Basic Labber integration
- Stretch
  - Linear classification
  - Report classifier distances
  - Full integration in lab measurement chain (run QST)

Work on Baseline

Baseline Completed
Work on Expected

Complete Expected
Begin Stretch

Work on Stretch

Wiggle/Debug Room

Now

Thanksgiving

12/12