# Quantum Computer Systems

## Research for Noisy Intermediate-Scale Quantum Computers

# Synthesis Lectures on Computer Architecture

*Synthesis Lectures on Computer Architecture* publishes 50- to 100-page publications on topics pertaining to the science and art of designing, analyzing, selecting and interconnecting hardware components to create computers that meet functional, performance and cost goals. The scope will largely follow the purview of premier computer architecture conferences, such as ISCA, HPCA, MICRO, and ASPLOS.

Quantum Computer Systems: Research for Noisy Intermediate-Scale Quantum Computers
Yongshan Ding and Frederic T. Chong
2020

Efficient Processing of Deep Neural Networks
Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S. Emer
2020

A Primer on Memory Consistency and Cache Coherence, Second Edition
Vijay Nagarajan, Daniel J. Sorin, Mark D. Hill, and David Wood
2020

Innovations in the Memory System
Rajeev Balasubramonian
2019

Cache Replacement Policies
Akanksha Jain and Calvin Lin
2019

The Datacenter as a Computer: Designing Warehouse-Scale Machines, Third Edition
Luiz André Barroso, Urs Hölzle, and Parthasarathy Ranganathan
2018

Principles of Secure Processor Architecture Design
Jakub Szefer
2018

General-Purpose Graphics Processor Architectures
Tor M. Aamodt, Wilson Wai Lun Fung, and Timothy G. Rogers
2018

Compiling Algorithms for Heterogenous Systems
Steven Bell, Jing Pu, James Hegarty, and Mark Horowitz
2018

Architectural and Operating System Support for Virtual Memory
Abhishek Bhattacharjee and Daniel Lustig
2017

Deep Learning for Computer Architects
Brandon Reagen, Robert Adolf, Paul Whatmough, Gu-Yeon Wei, and David Brooks
2017

On-Chip Networks, Second Edition
Natalie Enright Jerger, Tushar Krishna, and Li-Shiuan Peh
2017

Space-Time Computing with Temporal Neural Networks
James E. Smith
2017

Hardware and Software Support for Virtualization
Edouard Bugnion, Jason Nieh, and Dan Tsafrir
2017

Datacenter Design and Management: A Computer Architect's Perspective
Benjamin C. Lee
2016

A Primer on Compression in the Memory Hierarchy
Somayeh Sardashti, Angelos Arelakis, Per Stenström, and David A. Wood
2015

Research Infrastructures for Hardware Accelerators
Yakun Sophia Shao and David Brooks
2015

Analyzing Analytics
Rajesh Bordawekar, Bob Blainey, and Ruchir Puri
2015

Customizable Computing
Yu-Ting Chen, Jason Cong, Michael Gill, Glenn Reinman, and Bingjun Xiao
2015

Die-stacking Architecture
Yuan Xie and Jishen Zhao
2015

Single-Instruction Multiple-Data Execution
Christopher J. Hughes
2015

Power-Efficient Computer Architectures: Recent Advances
Magnus Själander, Margaret Martonosi, and Stefanos Kaxiras
2014

FPGA-Accelerated Simulation of Computer Systems
Hari Angepat, Derek Chiou, Eric S. Chung, and James C. Hoe
2014

A Primer on Hardware Prefetching
Babak Falsafi and Thomas F. Wenisch
2014

On-Chip Photonic Interconnects: A Computer Architect's Perspective
Christopher J. Nitta, Matthew K. Farrens, and Venkatesh Akella
2013

Optimization and Mathematical Modeling in Computer Architecture
Tony Nowatzki, Michael Ferris, Karthikeyan Sankaralingam, Cristian Estan, Nilay Vaish, and David Wood
2013

Security Basics for Computer Architects
Ruby B. Lee
2013

The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines, Second Edition
Luiz André Barroso, Jimmy Clidaras, and Urs Hölzle
2013

Shared-Memory Synchronization
Michael L. Scott
2013

Resilient Architecture Design for Voltage Variation
Vijay Janapa Reddi and Meeta Sharma Gupta
2013

Multithreading Architecture
Mario Nemirovsky and Dean M. Tullsen
2013

Performance Analysis and Tuning for General Purpose Graphics Processing Units (GPGPU)
Hyesoon Kim, Richard Vuduc, Sara Baghsorkhi, Jee Choi, and Wen-mei Hwu
2012

Automatic Parallelization: An Overview of Fundamental Compiler Techniques
Samuel P. Midkiff
2012

Phase Change Memory: From Devices to Systems
Moinuddin K. Qureshi, Sudhanva Gurumurthi, and Bipin Rajendran
2011

Multi-Core Cache Hierarchies
Rajeev Balasubramonian, Norman P. Jouppi, and Naveen Muralimanohar
2011

A Primer on Memory Consistency and Cache Coherence
Daniel J. Sorin, Mark D. Hill, and David A. Wood
2011

Dynamic Binary Modification: Tools, Techniques, and Applications
Kim Hazelwood
2011

Quantum Computing for Computer Architects, Second Edition
Tzvetan S. Metodi, Arvin I. Faruque, and Frederic T. Chong
2011

High Performance Datacenter Networks: Architectures, Algorithms, and Opportunities
Dennis Abts and John Kim
2011

Processor Microarchitecture: An Implementation Perspective
Antonio González, Fernando Latorre, and Grigorios Magklis
2010

Transactional Memory, Second Edition
Tim Harris, James Larus, and Ravi Rajwar
2010

Computer Architecture Performance Evaluation Methods
Lieven Eeckhout
2010

Introduction to Reconfigurable Supercomputing
Marco Lanzagorta, Stephen Bique, and Robert Rosenberg
2009

On-Chip Networks
Natalie Enright Jerger and Li-Shiuan Peh
2009

The Memory System: You Can't Avoid It, You Can't Ignore It, You Can't Fake It
Bruce Jacob
2009

Fault Tolerant Computer Architecture
Daniel J. Sorin
2009

The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines
Luiz André Barroso and Urs Hölzle
2009

Computer Architecture Techniques for Power-Efficiency
Stefanos Kaxiras and Margaret Martonosi
2008

Chip Multiprocessor Architecture: Techniques to Improve Throughput and Latency
Kunle Olukotun, Lance Hammond, and James Laudon
2007

Transactional Memory
James R. Larus and Ravi Rajwar
2006

Quantum Computing for Computer Architects
Tzvetan S. Metodi and Frederic T. Chong
2006

Quantum Computer Systems: Research for Noisy Intermediate-Scale Quantum Computers
Yongshan Ding and Frederic T. Chong

ISBN: 978-3-031-00637-1 paperback
ISBN: 978-3-031-01765-0 ebook
ISBN: 978-3-031-00062-1 hardcover

DOI 10.1007/978-3-031-01765-0

A Publication in the Springer series
*SYNTHESIS LECTURES ON COMPUTER ARCHITECTURE*

Lecture #51
Series Editors: Natalie Enright Jerger, *University of Toronto*
Margaret Martonosi, *Princeton University*
Founding Editor Emeritus: Mark D. Hill, *University of Wisconsin, Madison*
Series ISSN
Print 1935-3235 Electronic 1935-3243

Cover Photo: A Noisy Intermediate-Scale Quantum (NISQ) Machine from David Schuster's laboratory at the University of Chicago.

# Quantum Computer Systems

## Research for Noisy Intermediate-Scale Quantum Computers

Yongshan Ding
University of Chicago

Frederic T. Chong
University of Chicago

*SYNTHESIS LECTURES ON COMPUTER ARCHITECTURE #51*

# ABSTRACT

This book targets computer scientists and engineers who are familiar with concepts in classical computer systems but are curious to learn the general architecture of quantum computing systems. It gives a concise presentation of this new paradigm of computing from a computer systems' point of view without assuming any background in quantum mechanics. As such, it is divided into two parts. The first part of the book provides a gentle overview on the fundamental principles of the quantum theory and their implications for computing. The second part is devoted to state-of-the-art research in designing practical quantum programs, building a scalable software systems stack, and controlling quantum hardware components. Most chapters end with a summary and an outlook for future directions. This book celebrates the remarkable progress that scientists across disciplines have made in the past decades and reveals what roles computer scientists and engineers can play to enable practical-scale quantum computing.

# KEYWORDS

quantum computing, computer architecture, quantum compilation, quantum programming languages, quantum algorithms, noise mitigation, error correction, qubit implementations, classical simulation

# Contents

**Preface** . . . . . . . . . . xv

**Acknowledgments** . . . . . . . . . . xvii

**List of Notations** . . . . . . . . . . xix

**PART I Building Blocks** . . . . . . . . . . 1

**1 Introduction** . . . . . . . . . . 3

1.1 The Birth of Quantum Computing . . . . . . . . . . 3
1.1.1 The Rise of a New Computing Paradigm . . . . . . . . . . 4
1.1.2 What Is a Quantum Computer? . . . . . . . . . . 5
1.2 Models of Quantum Computation . . . . . . . . . . 5
1.2.1 Analog Model . . . . . . . . . . 6
1.2.2 Gate-Based Model . . . . . . . . . . 6
1.2.3 Measurement-Based Model . . . . . . . . . . 6
1.3 A QPU for Classical Computing . . . . . . . . . . 7
1.3.1 Architectural Design of a QPU . . . . . . . . . . 8
1.4 Quantum Technologies . . . . . . . . . . 9
1.5 A Road Map for Quantum Computers . . . . . . . . . . 10
1.5.1 Computer Science Research Opportunities . . . . . . . . . . 11

**2 Think Quantumly About Computing** . . . . . . . . . . 13

2.1 Bits vs. Qubits . . . . . . . . . . 13
2.1.1 Computing with Bits: Boolean Circuits . . . . . . . . . . 13
2.1.2 Computing with Qubits: Quantum Circuits . . . . . . . . . . 20
2.1.3 Architectural Constraints of a Quantum Computer . . . . . . . . . . 25
2.2 Basic Principles of Quantum Computation . . . . . . . . . . 26
2.2.1 Quantum States . . . . . . . . . . 27
2.2.2 Composition of Quantum Systems . . . . . . . . . . 28

2.2.3 Measurements . . . . . 31
2.2.4 Quantum Gates . . . . . 33
2.3 Noisy Quantum Systems . . . . . 37
2.3.1 Quantum Probability . . . . . 37
2.3.2 Operator Sum Representation . . . . . 39
2.3.3 Qubit Decoherence and Gate Noise . . . . . 41
2.4 Qubit Technologies . . . . . 42
2.4.1 Trapped Ion Qubits . . . . . 43
2.4.2 Superconducting Qubits . . . . . 48
2.4.3 Other Promising Implementations . . . . . 53

**3 Quantum Application Design . . . . . 55**
3.1 General Features . . . . . 55
3.1.1 The Computing Process . . . . . 56
3.1.2 The Query Model and Quantum Parallelism . . . . . 56
3.1.3 Complexity, Fidelity, and Beyond . . . . . 59
3.2 Gate-Based Quantum Algorithms . . . . . 62
3.2.1 Deutsch–Josza Algorithm . . . . . 63
3.2.2 Bernstein–Vazirani Algorithm . . . . . 65
3.3 NISQ Quantum Algorithms . . . . . 67
3.3.1 Variational Quantum Eigensolver (VQE) . . . . . 67
3.3.2 Quantum Approximate Optimization Algorithm (QAOA) . . . . . 68
3.4 Summary and Outlook . . . . . 69

**PART II Quantum Computer Systems . . . . . 71**

**4 Optimizing Quantum Systems–An Overview . . . . . 73**
4.1 Structure of Quantum Computer Systems . . . . . 73
4.2 Quantum-Classical Co-Processing . . . . . 75
4.3 Quantum Compiling . . . . . 76
4.4 NISQ vs. FT Machines . . . . . 78

**5 Quantum Programming Languages . . . . . 81**
5.1 Low-Level Machine Languages . . . . . 81
5.2 High-Level Programming Languages . . . . . 82

5.3 Program Debugging and Verification . . . 83
  5.3.1 Tracing via Classical Simulation . . . 84
  5.3.2 Assertion via Quantum Property Testing . . . 85
  5.3.3 Proofs via Formal Verification . . . 89
5.4 Summary and Outlook . . . 89

**6 Circuit Synthesis and Compilation . . . 91**
6.1 Synthesizing Quantum Circuits . . . 91
  6.1.1 Choice of Universal Instruction Set . . . 92
  6.1.2 Exact Synthesis . . . 94
  6.1.3 Approximate Synthesis . . . 102
  6.1.4 Higher-Dimensional Synthesis . . . 104
6.2 Classical vs. Quantum Compiler Optimization . . . 105
6.3 Gate Scheduling and Parallelism . . . 106
  6.3.1 Primary Constraints in Scheduling . . . 108
  6.3.2 Scheduling Strategies . . . 112
  6.3.3 Highlight: Gate Teleportation . . . 113
6.4 Qubit Mapping and Reuse . . . 115
  6.4.1 Finding a Good Qubit Mapping . . . 117
  6.4.2 Strategically Reusing Qubits . . . 119
  6.4.3 Highlight: Uncomputation . . . 120
6.5 Summary and Outlook . . . 124

**7 Microarchitecture and Pulse Compilation . . . 127**
7.1 From Gates to Pulses–An Overview . . . 127
  7.1.1 General Pulse Compilation Flow . . . 127
7.2 Quantum Controls and Pulse Shaping . . . 128
  7.2.1 Open-Loop vs. Closed-Loop Control . . . 129
7.3 Quantum Optimal Control . . . 130
  7.3.1 Highlight: Compilation for Variational Algorithms . . . 131
7.4 Summary and Outlook . . . 132

**8 Noise Mitigation and Error Correction . . . 133**
8.1 Characterizing Realistic Noises . . . 133
  8.1.1 Measurements of Decoherence . . . 135
  8.1.2 Quantum-State Tomography . . . 136
  8.1.3 Randomized Benchmarking . . . 136

8.2 Noise Mitigation Strategies . . . 137
    8.2.1 Randomized Compiling . . . 138
    8.2.2 Noise-Aware Mapping . . . 140
    8.2.3 Crosstalk-Aware Scheduling . . . 140
8.3 Quantum Error Correction . . . 141
    8.3.1 Basic Principles of QEC . . . 141
    8.3.2 Stabilizer Codes . . . 145
    8.3.3 Transversality and Eastin–Knill Theorem . . . 146
    8.3.4 Knill's Error Correction Picture . . . 147
8.4 Summary and Outlook . . . 149

**9 Classical Simulation of Quantum Computation . . . 151**
9.1 Strong vs. Weak Simulation: An Overview . . . 151
    9.1.1 Distance Measures . . . 153
9.2 Density Matrices: The Schrödinger Picture . . . 154
9.3 Stabilizer Formalism: The Heisenberg Picture . . . 156
9.4 Graphical Models and Tensor Network . . . 159
9.5 Summary and Outlook . . . 163

**10 Concluding Remarks . . . 165**

**Bibliography . . . 169**

**Authors' Biographies . . . 203**

# Preface

Quantum computing is at a historic time in its development and there is a great need for research in quantum computer systems. This book stems from a course we co-taught in 2018 and the research efforts of the EPiQC NSF Expedition in Computing and others. Our goal is to provide a broad overview of some of the emerging research areas in the development of practical computing systems based upon emerging noisy intermediate-scale quantum hardware. It is our hope that this book will encourage other researchers in the computer systems community to pursue some of these directions and help accelerate real-world applications of quantum computing.

Despite the impressive capability of today's digital computers, there are still some computational tasks that are beyond their reach. Remarkably, some of those tasks seem to be relatively easy with a quantum computer. Over the past four decades or so, our understanding of the theoretical power of quantum and skills in quantum engineering has advanced significantly. Small-scale prototypes of progammable quantum computers are emerging from academic and industry labs around the world. This is undoubtedly an exciting time, as we may be soon fortunate enough to be among the first to witness the application of quantum computers on problems that are unfeasible for today's classical computers. What has been truly remarkable is that the field of quantum information science has brought scientists together across disciplines—physicists, electrical engineers, computer architects, and theorists, just to name a few.

Looking back at the historical progress in digital computers, we remark upon the three major milestones that led to the integration of millions of computational units that make up the computing power in today's computers: low-cost integrated circuit technology, efficient architectural design, and interconnected software ecosystem. It is not too unrealistic to assume that the evolution of quantum computers will follow a similar trajectory; we are starting to see some innovations in hardware, software, and architecture designs that have the potential to scale up well. The progress and prospect of the new paradigm of computing has motivated us to write this Synthesis Lecture, which hopefully can bring together more and more computer scientists and engineers to join the expedition to practical-scale quantum computation.

This introduction to quantum computer systems should primarily appeal to computer systems researchers, software engineers, and electrical engineers. The focus of this book is on systems research for noisy intermediate-scale quantum (NISQ) computers, highlighting the recent progress and addressing the near-term challenges for realizing the computational power of QC systems.

## Reading This Book

The aim of this book is to provide computer systems researchers and engineers with an introductory guide to the general principles and challenges in designing practical quantum computing systems. Compared to its predecessor in the series, *Quantum Computing for Computer Architects* by Metodi, Faruque, and Chong [1], this book targets near-term progress and prospects of quantum computing. Throughout the book, we emphasize how computer systems researchers can contributes to the exciting emerging field. As such, the structure of this book is as follows. Chapter 2 reviews the central concepts in quantum computation, compares and contrasts with those of classical computation, and discusses the leading technologies for implementing qubits. Chapter 3 summarizes the general features in quantum algorithms and reviews some of the important NISQ applications.

The second part of the book starts in Chapter 4 with an overview of the quantum architectural vertical stack and the cross-cutting themes that enable synergy among the different disciplines in the field. The rest of the book illuminates the opportunities in quantum computer systems research, broadly split into five tracks: (i) Chapter 5 describes existing quantum programming languages and techniques for debugging and verification; (ii) Chapter 6 introduces important quantum compilation methods including circuit optimization and synthesis; (iii) Chapter 7 dives into low-level quantum controls, pulse generation, and calibration; (iv) a number of noise mitigation and error correction techniques are reviewed in Chapter 8; (v) Chapter 9 discusses different methods in classical simulations of quantum circuits and their implications; and (vi) a summary of progress and prospects of quantum computer systems research can be found in Chapter 10.

The reader is encouraged to start with the Summary and Outlook section in some chapters for a quick overview of fundamental concepts, highlights of state-of-the-art research, and discussions of future directions.

Yongshan Ding and Frederic T. Chong
Chicago, June 2020

# Acknowledgments

Our views in the book are strongly informed by ideas formed from discussions with Yuri Alexeev, Kenneth Brown, Chris Chamberland, Isaac Chuang, Andrew Cross, Bill Fefferman, Diana Franklin, Alexey Gorshkov, Hartmut Haeffner, Danielle Harlow, Aram Harrow, Henry Hoffman, Andrew Houck, Ali Javadi-Abhari, Jungsang Kim, Peter J. Love, Margaret Martonosi, Akimasa Miyake, Chris Monroe, William Oliver, John Reppy, David Schuster, Peter Shor, Martin Suchara, members of the EPiQC Project (Enabling Practical-scale Quantum Computation, an NSF Expedition in Computing), and members of the STAQ Project (Software-Tailored Architecture for Quantum co-design). Thanks are extended to the students who took the 2018 course on quantum computer systems for their helpful lecture scribing notes: Anil Bilgin, Xiaofeng Dong, Shankar G. Menon, Jean Salac, and Lefan Zhang, among others.

Thanks to Morgan & Claypool Publishers for making the publication this book possible. Many thanks to Michael Morgan, who invited us to write on the subject, for his patience and encouragement. Thanks also to our Synthesis Lecture series editors Natalie Enright Jerger and Margaret Martonosi, who shepherded this project to its final product. YD and FTC are grateful to Frank Mueller and the anonymous reviewers for providing in-depth comments and suggestions on the original manuscript. Thanks to Sara Kreiman for her thorough copyedit of the book.

YD has learned a tremendous amount from his advisor FTC, and is very grateful for FTC's mentorship in quantum information science research and education. YD also thanks Ryan O'Donnell, who first introduced him to the field of quantum computation and information. YD worked on this book while visiting the Massachusetts Institute of Technology. YD especially thanks Isaac Chuang, Aram Harrow, and Peter Shor for the many inspiring discussions during his visit. YD thanks all of his colleagues, friends, and relatives for their encouragement and support in writing and finishing the book, especially Meizi Liu, and YD's parents, Genlin Ding and Shuowen Feng.

Finally, YD and FTC gratefully acknowledge the support from the National Science Foundation, specifically by EPiQC, an NSF Expedition in Computing, under grants CCF-1730449, in part by STAQ, under grant NSF Phy-1818914, and in part by DOE grants DE-SC0020289 and DE-SC0020331.

Yongshan Ding and Frederic T. Chong
Chicago, June 2020

# List of Notations

The nomenclature and notations used in this book may be unfamiliar to many readers and may have different meanings in a different context. We devote this section to clarifying some of the conventions this book uses to prevent confusion.

## Systems Terminology

- *Adiabatic quantum computing* is a model of analog quantum computing where a quantum system remains in the ground state energy.
- *Analog quantum computing (AQC)* is a model of quantum computation such that the state of a quantum system is evolved smoothly.
- *Boolean circuit* is a model of classical computation that expresses computation by sending data through a combination of logic gates.
- *FT* refers to being fault tolerant; a fault-tolerant quantum computer relies on quantum error correction.
- The *gate scheduling* problem is to design an ordering or synchronization of quantum gates to be applied to the qubits in the target architecture, under constraints such as data dependencies, parallelism, communication, and noise.
- *Hamiltonian* refers to the mathematical representation of the energy configuration of a physical system. It is commonly used as a linear algebraic operator in quantum mechanics.
- *Host processor* is an abstraction that refers to the classical computer that controls the processes in quantum computer systems.
- *Quantum annealing* is a model of analog quantum computing where-in the quantum systems interact with the thermal environment.
- *Lambda calculus* is a model of classical computation based on functional expressions using variable binding and substitution.
- *Measurement-based quantum computing* (MBQC) is a model of computation that performs computation via only measurements on qubits previously initialized to a cluster state.

- A *NISQ* computer refers to a noisy intermediate-scale quantum computer.
- *Turing machine* is a model of classical computation for abstract computing machines based on manipulating data sequentially on a strip of tape following a set of rules.
- *Quantum compiling* refers to the framework for efficiently implementing a given quantum program or target unitary to high precision, using gates from a set of primitive instructions supported in the underlying quantum architecture.
- *Quantum communication* is a branch of quantum technology where-in entangled qubits are used to encrypt and transmit data.
- *Quantum circuit synthesis* refers to the technique that constructs a gate out of a series of primitive operation.
- *Quantum device topology* (or device connectivity) describes the layout of the physical qubits and the allowed direct interactions between any pair of qubits.
- *Quantum logic gates* (or *qubit operations* or *quantum instructions*) are transformations to be applied to qubits, represented by unitary matrices.
- The *qubit mapping* problem aims to find an optimal mapping from the qubit registers in a quantum program to the qubits in the target architecture, under constraints such as system size, data dependencies, communication, ancilla reuse, and noise.
- *Quantum processing unit (QPU)* refers to a hardware component that implements qubits as well as the control apparatus.
- A *quantum program* is an abstraction that refers to the sequence of instructions and control flow that a quantum computer must follow according to a protocol or an algorithm.
- *Quantum sensing* is a branch of quantum technology that takes advantage of quantum coherence to perform measurements of physical quantities.
- *Quantum simulation* is a branch of quantum technology that studies the structures and properties of electronic or molecular systems.
- *Schoelkopf's law* is an empirical scaling projection for quantum decoherence—delayed by a factor of 10 roughly every three years.
- The *von Neumann architecture* is a stored-program computer architecture that controls instruction fetch and data operations via a common system computer bus.

## Linear Algebra and Probability in Quantum Computing

- The *basis* of a qubit is a set of linearly independent vectors that span the Hilbert space. The two most common bases for single qubits are the computational basis ($z$ basis):

  $$\{|0\rangle, |1\rangle\} \equiv \left\{\begin{pmatrix}1\\0\end{pmatrix}, \begin{pmatrix}0\\1\end{pmatrix}\right\},$$

  and the Fourier basis ($x$ basis):

  $$\{|+\rangle, |-\rangle\} \equiv \left\{\begin{pmatrix}1/\sqrt{2}\\1/\sqrt{2}\end{pmatrix}, \begin{pmatrix}1/\sqrt{2}\\-1/\sqrt{2}\end{pmatrix}\right\}.$$

- The *Bloch sphere* is a visualization of single-qubit Hilbert space $\mathcal{H}$ in three-dimensional Euclidean space $\mathbb{R}^3$:

  $$\rho(x, y, z) = \frac{1}{2}(I + x\sigma_x + y\sigma_y + z\sigma_z).$$

- The *bra vector* is the conjugate transpose of a ket vector:

  $$\langle\psi| = \begin{pmatrix}\alpha^* & \beta^*\end{pmatrix}.$$

- A *cluster state* is a quantum state defined by a graph, where the nodes in the graph are qubits initialized to $|+\rangle$ state, and the edges are controlled-Z gates between the qubits.

- A *complex number* $z \in \mathbb{C}$ is a number in the form of $a + bi$, where $a, b$ are real numbers and $i$ is an imaginary unit satisfying $i^2 = -1$. $a$ is called the read part, and $b$ is called the imaginary part of $z$. The *conjugate* of $z$ is $z^* = a - bi$.

- The *conjugate transpose* of a matrix $M$ is denoted as $M^\dagger$ whose matrix elements are:

  $$[M^\dagger]_{ij} = [M]_j i^*.$$

- An *EPR* pair refers to two qubits in the quantum state $|epr\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$.

- The common mapping is $|0\rangle$ for *ground energy state*, and $|1\rangle$ for *first excited energy state*. In the context of the physical implementation of a qubit, the computational basis corresponds to the discrete energy levels.

- A complex square matrix is *Hermitian* if its complex conjugate transpose $H^\dagger$ is equal to itself:

  $$H^\dagger = H.$$

- The *Hilbert space* $\mathcal{H}$ is complex inner product space in which a $n$-qubit quantum state is a $2^n$-dimensional vector of complex entries.
- The *inner product* of two quantum states $|\psi\rangle = \sum_j \alpha_j |j\rangle, |\phi\rangle = \sum_k \beta_k |k\rangle$ is $\langle\psi|\phi\rangle = \sum_i = \alpha_i^* \beta_i$.
- An *identity matrix* $I$ is a matrix with 1 along the diagonal and 0 everywhere else.
- For any real number $p \geq 1$, the $\ell_p$ *norm* of a vector $\mathbf{x} = (x_1, \ldots, x_n)$ is defined as

$$||\mathbf{x}||_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}.$$

- $e^M$ and $\exp(M)$ are notations for *matrix exponential*, which is defined as:

$$e^M = \sum_{k=0}^{\infty} \frac{1}{k!} M^k.$$

- A *mixed quantum state* or *density matrix* is a probability ensemble of pure quantum states: $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$.
- The *Pauli matrices* are

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

- A *probability distribution* refers to a finite set of non-negative real numbers $p_i$ that sums to 1: $p_i \geq 0$ and $\sum_i p_i = 1$.
- A *quantum channel* is a linear mapping from one mixed state to another mixed state

$$\rho \to \mathcal{E}(\rho).$$

- *Quantum states* are represented by (column) vectors in the Hilbert space using Dirac's *ket vector* notation:

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}.$$

- *sgn*$(x)$ is the sign of the number $x$.
- The *tensor product* of two quantum states $|\psi\rangle = \sum_j \alpha_j |j\rangle, |\phi\rangle = \sum_k \beta_k |k\rangle$ is $|\psi\rangle \otimes |\phi\rangle = \sum_{j,k} = \alpha_j \beta_k (|j\rangle \otimes |k\rangle)$.

- The *trace* of a matrix $A$ is the sum of its diagonal elements, $tr(A) = \sum_i A_{ii} = \sum_i \langle e_i|A|e_i\rangle$, where $|e_i\rangle$ is the basis vector with 1 at the $i^{th}$ index and 0 everywhere else.

- A complex square matrix $U$ is *unitary* if its complex conjugate transpose $U^\dagger$ is also its inverse:

$$U^\dagger U = UU^\dagger = I.$$

- The system, or *wave function*, of a qubit can be written as a linear combination of basis states.