Hardware-Aware Quantum Education: Six Modules Bridging Theory, Hardware, and Software for Next-Generation Quantum Workforce

Shiva Dahal[1], John T. Yi[1], and Jinsuk Baek[2*]
[1] Department of Chemistry, Winston-Salem State University
Winston-Salem, NC 27110, USA
[e-mail: dahals@wssu.edu, yijt@wssu.edu]
[2] Department of Computer Science, Winston-Salem State University
Winston-Salem, NC 27110, USA
[e-mail: baekj@wssu.edu]
[*]Corresponding author: Jinsuk Baek

Received October 14, 2025; revised November 23, 2025; revised December 1, 2025; accepted December 11, 2025; published May 31, 2026

Abstract

Quantum computing promises solutions to problems that are beyond the reach of classical machines; however, current noisy intermediate-scale quantum (NISQ) devices require both theoretical insight and hardware expertise. This paper presents a six-module curriculum designed to bridge that gap for undergraduates from diverse disciplines. The modules progress from Python programming to electronics and circuits, vacuum technology, cryogenic measurements, microwave and RF systems, and culminate with quantum simulations and computations. Implemented at Winston-Salem State University with thirty students across computer science, information technology, mathematics, chemistry, physics, and biology, the program combined lectures, group projects, and presentations. Hands-on activities included resistor network analysis, magnetic levitation, vacuum chamber assembly, cryogenics measurements, RF component analysis, and entanglement experiments, enabling students to connect abstract principles with laboratory practice. Outcomes showed increased confidence, collaboration, critical thinking, and skill set building, even among those with no prior background. Students advanced to research-oriented projects like secure quantum communication protocols and the Quantum Mini Tennis game, which explored gate operations and qubit transformations, demonstrating that hardware-aware quantum education lowers barriers, fosters interdisciplinary participation, and prepares students for NISQ-era challenges.

Keywords: Hardware-aware quantum education, Quantum computing curriculum, Noisy intermediate-scale quantum (NISQ), Qiskit and IBM Quantum, Secure quantum communication

http://doi.org/10.3837/tiis.2026.05.019 **ISSN : 1976-7277**

1. Introduction

Quantum computing is rapidly emerging as a transformative paradigm capable of addressing computational challenges that are intractable for classical systems, including certain NP-complete and NP-hard problems. Despite this potential, today's quantum devices remain constrained by instability, short coherence times, and high error rates, making it difficult to execute complex algorithms reliably. These challenges underscore the critical gap between the theoretical foundations of quantum mechanics and the practical realization of quantum computation on physical hardware [1, 2].

In classical computing, software developers typically build applications without requiring detailed knowledge of the underlying hardware. High-level programming languages, compilers, and device drivers abstract away the complexity of circuits and microarchitecture, enabling programmers to focus exclusively on algorithms. Consider the following example of a simple assignment statement in a high-level programming language:

A = A + B;

From the programmer's perspective, this line adds two numbers and stores the result. However, its execution involves multiple hardware-level steps. The CPU must:

1. Load the values of A and B from memory into registers.
2. Decode the instruction and route it through the arithmetic and logic unit (ALU).
3. Perform the binary addition using a full-adder logic circuit that processes inputs X, Y, and a carry-in (C_{in}) to produce a sum (S) and a carry-out (C_{out}).
4. Store the result back into memory at the destination address of A.

This process, shown schematically in **Fig. 1**, demonstrates how even a trivial line of code is deeply tied to the structure of the hardware. Most classical programmers do not need to be aware of these details, as the abstraction layers handle them.

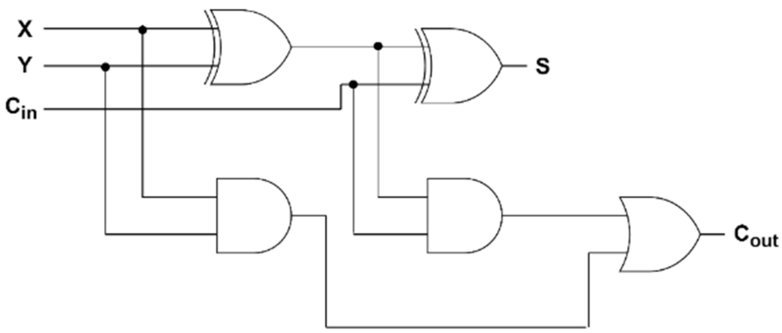


Fig. 1. Logic circuit for a full adder with carry implementing S = X + Y

By contrast, quantum computing imposes fundamentally different demands. Developing quantum algorithms requires at least a basic understanding of qubit initialization, gate operations, decoherence mechanisms, and measurement constraints. Without such hardware awareness, students and researchers often struggle to interpret experimental results or to design software that can function reliably under the limitations of noisy intermediate-scale quantum (NISQ) devices [3, 4].

The coupling between the development of quantum algorithms and knowledge of the underlying physical processes is crucial as a guide to realizing qubit operations in hardware. Unlike classical computations, in which layers of abstraction can mask the behavior of individual transistors, NISQ devices can make the cloaking properties of software ineffective at hiding deeply embedded information. For qubits, this initially combines thermal relaxation and pulse calibration precision during qubit initialization, as well as single- and two-qubit gates with microwave control pulses, flux control or cross resonance interaction, each of them contributing specific error channels to the device. Noise sources such as energy relaxation (T_1), dephasing (T_2), gate infidelity, and qubit-qubit crosstalk alter the intended evolution of quantum states and typically cause output distributions that are far from ideal simulation. Readout resonators, amplifiers and threshold-based discrimination also contribute to measurement errors. Because these compounding factors are the ultimate drivers of computational outcomes, students must develop an appreciation for the limitations imposed by actual hardware to misinterpret deviations with respect to theory, explore noise behavior and design circuits with which one can expect reliable operation under NISQ-era constraints. This puts pressure for a curriculum that is highly based on hardware understanding.

To address this challenge, we developed and implemented a six-module hardware-aware introductory quantum education program at Winston-Salem State University (WSSU). The program is designed to lower the barrier of entry for students encountering quantum science for the first time, while equipping them with the necessary knowledge to bridge the gap between theory and practice. This paper presents these modules in a tutorial format, with step-by-step explanations and experimental demonstrations, to highlight their educational impact and contribution to building a diverse and capable quantum workforce.

2. Why Hardware-Aware Quantum Computing?

Quantum computing differs fundamentally from classical computing not only in theory but also in practice. In digital systems, error handling has been extensively developed through decades of engineering progress. Techniques such as checksums, Hamming codes, and cyclic redundancy checks enable reliable data transmission and computation by adding redundancy and correcting bit-level errors. In these systems, software developers can generally assume the existence of robust hardware and rely on abstraction layers that shield them from device-level failures.

By contrast, quantum systems face unique challenges that invalidate these traditional strategies. One fundamental barrier is the no-cloning theorem, which prohibits copying an unknown quantum state. As a result, qubits cannot be redundantly duplicated for straightforward error detection and correction. Instead, error correction in quantum computing requires encoding a logical qubit into multiple physical qubits. For instance, state-of-the-art surface code implementations require as many as 17 physical qubits to protect a single logical qubit from one error, and 48 or more physical qubits for correcting two-bit errors.

Environmental noise further exacerbates this fragility. Quantum states are highly sensitive to disturbances from temperature fluctuations, stray electromagnetic fields, radiation, and imperfect shielding. Even slight environmental variations can induce decoherence or collapse quantum superpositions prematurely.

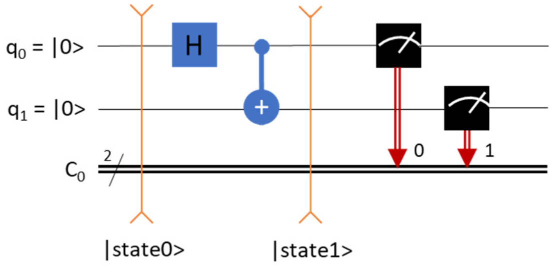


Fig. 2. Quantum circuit for generating a Bell state entanglement

A simple but illustrative example is the creation of a Bell state using two qubits. Let us develop a simple quantum software that creates one of the Bell states with two-qubit entanglement, as shown in **Fig. 2**. Both qubits, q0 and q1, are initially in the |0⟩ state. Since multiple qubits are represented using the tensor product ⊗, the initial state can be written as:

$$|\text{state0}\rangle = |0\rangle \otimes |0\rangle = |00\rangle \tag{1}$$

To generate entanglement, we first apply a Hadamard gate (H) to qubit q0, creating a superposition, and then apply a CNOT gate with q0 as the control and q1 as the target. Mathematically, this process evolves as:

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{2}$$

$$\begin{aligned} |\text{state1}\rangle &= \text{CNOT}_{1,2}\, H_1\, |\text{state0}\rangle = \text{CNOT}_{1,2}\, H_1\, (|0\rangle \otimes |0\rangle) \\ &= \text{CNOT}_{1,2}\, (H|0\rangle \otimes |0\rangle) = \text{CNOT}_{1,2}\, \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |0\rangle\right) \\ &= \text{CNOT}\, \left(\frac{|00\rangle + |10\rangle}{\sqrt{2}}\right) = \frac{|00\rangle + |11\rangle}{\sqrt{2}} \end{aligned} \tag{3}$$

This final state is the maximally entangled Bell state, often denoted as $|\Phi^+\rangle$. Ideally, measurement outcomes are evenly split between |00⟩ and |11⟩.

However, when this same circuit is implemented on real quantum hardware, deviations from the theoretical distribution are observed. As discussed later in Section 3.6, the hardware results reveal small but non-negligible populations in the |01⟩ and |10⟩ states due to gate imperfections, measurement noise, and environmental interference, consistent with recent experimental findings on NISQ-era devices [2, 5]. This example highlights the crucial need for hardware-awareness in introductory quantum education. Without knowledge of how decoherence, noise, and device imperfections affect outcomes, students may fail to reconcile discrepancies between theory and practice. A hardware-aware perspective ensures that learners can interpret such results accurately, preparing them to engage productively in the noisy intermediate-scale quantum (NISQ) era. This perspective forms the foundation of our quantum education program. By embedding hardware-awareness into each module—ranging from electronics and circuits to vacuum technology, cryogenics, and quantum simulations—we provide students with a holistic understanding of quantum systems. This approach equips them with not only computational skills but also the experimental intuition required to interpret results, troubleshoot errors, and contribute effectively to the advancement of quantum science and engineering.

3. Tutorial Modules for Hardware-Aware Quantum Education

The six modules of our program were implemented with a cohort of thirty undergraduate students. About half of the participants were Computer Science or Information Technology majors, while the other half represented diverse STEM backgrounds, including Mathematics, Chemistry, Physics, and Biology. Most students were freshmen or sophomores, bringing varied levels of prior exposure to programming and laboratory work. To foster collaboration and hands-on learning, students were divided into six groups. Each module followed a four-week cycle:

- Weeks 1–2: Topic-focused lectures and interactive hands-on exercises
- Week 3: Group projects to expand knowledge and strengthen skills
- Week 4: Development of group presentations with practice in delivery and Q&A

This structure supported both individual skill-building and teamwork, helping students apply newly learned concepts to practical tasks. The six modules provide a broad introduction to the essential tools and experimental methods of quantum science. Each module combines theory with hands-on laboratory work, allowing students to link abstract concepts to real-world applications.

Although each module uses a four-week structure, this timeline functions as a scaffold rather than a self-contained unit. Skills introduced early are reinforced through group projects and extended work in open-lab sessions supervised by faculty and a postdoctoral researcher. Because the modules build on one another, students gradually develop deeper technical competence as they apply earlier knowledge to more advanced experiments.

The modules are as follows:

- Python Programming: Introduces Python fundamentals and guides students through Qiskit exercises, preparing them to implement quantum algorithms on simulators and real quantum hardware.
- Electronics and Circuits: Covers basic electronic components and circuit building, including classical logic gates, and explains their relevance to controlling and measuring quantum devices.
- Vacuum Technology: Teaches the operation of vacuum pumps, gauges, and leak-detection tools, emphasizing the role of vacuum environments in isolating quantum systems from external noise.
- Cryogenic Measurements: Introduces superconducting materials, low-temperature physics, and cryostat operation, highlighting why cryogenic conditions are essential for superconducting qubits.
- Microwave (MW) and Radiofrequency (RF) Systems: Covers basic signal concepts and the use of MW/RF components for tuning quantum states, performing gate operations, and characterizing quantum devices.
- Quantum Simulations and Computations: Integrates all previous modules through hands-on design, simulation, and execution of quantum circuits, with analysis of noise effects and error impacts in real NISQ hardware.

Table 1. Objectives and outcome of different modules

Modules	Objectives	Outcome
1. Python Programming	• Learn Fundamental Coding • Hands-on Exercise with Qiskit	Students gained familiarity with Python programming and utilized it in the quantum simulation module.
2. Electronics and Circuits	• Learn fundamental knowledge of electronic components • Build basic circuits • Learn and build basic classical gates	Basic knowledge about circuits and components was understood, and classic circuits and gates were constructed.
3. Vacuum Technology	• Learn about the vacuum system • Learn about vacuum components • Hands-on training on vacuum chamber set-up and pumping	Students gained an understanding of the different components and hands-on training in operating a vacuum system.
4. Cryogenics Measurements	• Learn about superconducting materials • Measure R vs T of different materials • Experiments in a cryogenic environment	Students learned about the properties of superconducting materials and the safety procedures when working at low temperatures.
5. MW and RF Systems	• Learn about signal basics • Learn about MW and RF components • Implementing those components to see how the signal is affected	Students understood basic signal terminology and how to modify or filter signals using a signal generator, oscilloscope, VNA, MW, and RF components.
6. Quantum Simulations and Computations	• Design and simulate quantum circuits • Learn about noise and error	Students designed, simulated, and executed the quantum circuits and analyzed the results.

The objectives and outcomes are summarized in **Table 1**. Together, these modules provide a holistic curriculum that not only builds technical expertise but also develops students' confidence in approaching interdisciplinary problems at the intersection of physics, engineering, and computer science.

By framing the modules as tutorials, our goal is to make quantum science more approachable to students from computing and information science backgrounds, many of whom lack prior exposure to advanced physics, while fostering diversity in the quantum workforce.

3.1 Python Programming

Python is introduced as the foundational programming language for this curriculum, chosen for its simplicity, readability, and widespread adoption in scientific computing. Because many students in our program enter without prior programming experience, Python provides an accessible starting point. The goal of this module is to build confidence in coding fundamentals before students progress to more advanced topics, including quantum simulations.

```python
# Basic output
print("Python Programming!")

# Variables and arithmetic
x = 7
y = 3
print("Sum:", x + y)

# Looping through a list
numbers = [1, 2, 3, 4, 5]
for n in numbers:
    print("Square:", n**2)
```

Fig. 3. Python code for variables, arithmetic, and loops

Students begin by learning how to declare variables, perform arithmetic operations, and use control structures such as loops. For example, a short script demonstrates how to print messages, add integers, and calculate the squares of numbers within a loop. This introductory exercise, illustrated in **Fig. 3**, enables students to observe how programming logic is translated into actual computation.

```python
import numpy as np

# Function definition
def multiply(a, b):
    return a * b

print("Multiplication:", multiply(4, 6))

# Using NumPy for array operations
arr = np.array([2, 4, 6, 8])
print("Array mean:", np.mean(arr))
print("Array standard deviation:", np.std(arr))
```

Fig. 4. Python code for functions and NumPy operations

Once comfortable with these basics, students are introduced to writing reusable code through functions and to using external libraries such as NumPy. NumPy is particularly valuable for quantum science, as it simplifies linear algebra operations that are central to quantum algorithms. A representative example is provided in **Fig. 4**, where students define a simple multiplication function and then apply NumPy functions to analyze an array of numbers.

Learning to install, import, and apply such libraries is also an intentional step in preparing students for the final module, where they will use the Qiskit library to design and execute quantum circuits on simulators and real hardware.

After completing these structured exercises, students are organized into small groups to work on collaborative projects. Typical projects include:

- Writing programs that compute basic statistics from a dataset
- Designing a simple text-based game using conditional statements
- Creating a script that processes lists of numerical data and outputs summary results.

These projects not only reinforce technical content but also foster problem-solving and teamwork, skills that are essential in scientific research environments. By the end of this module, students with little or no programming background are able to write functional Python programs, use external libraries, and collaborate effectively with peers. This foundation equips them for later modules, where Python will serve as the bridge to quantum-specific tools and experiments.

3.2 Electronics and Circuits

Classical circuits are the foundation of modern technology, from consumer electronics to high-performance computing. They also underpin digital logic and computation, which are essential for controlling and interpreting quantum states. A strong grasp of classical electronics is therefore critical for students preparing to work with quantum hardware.

This module begins with an introduction to voltage, defined as the work required to move an electrical charge:

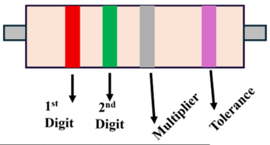
$$V = \frac{E}{Q} \tag{4}$$

where V is voltage (volts), E is energy (joules), and Q is charge (coulombs). Students measure the voltage of a commercial battery using a digital multimeter to ground this concept in practice.

To strengthen their understanding of resistance, students are introduced to identifying resistor values (in ohms) and tolerances using color codes [6], which provide a quick and standardized method for determining resistance without measurement instruments.

In a typical 4-band resistor, the first two bands represent significant digits, the third band indicates the multiplier, and the fourth band specifies the tolerance. (**Fig. 5(a)**). For example:

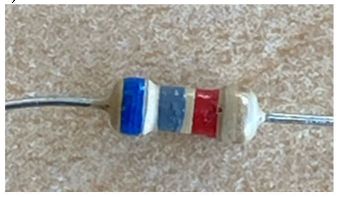
- A resistor with bands brown–black–yellow-silver corresponds to $10\times10^4 \pm10\%\Omega$ Ω = 100 kΩ with a ±10% tolerance. (**Fig. 5(b)**)
- A resistor with bands blue-grey-red-silver corresponds to $68\times10^2 \pm10\%\Omega$ = 6.8 kΩ with a ±10% tolerance. (**Fig. 5(c)**)



Black	0	0	x 1	
Brown	1	1	x 10	±1%
Red	2	2	x 10^2	±2%
Orange	3	3	x 10^3	±3%
Yellow	4	4	x 10^4	±4%
Green	5	5	x 10^5	±0.5%
Blue	6	6	x 10^6	±0.25%
Violent	7	7	x 10^7	±0.1%
Grey	8	8	x 10^8	±0.05%
White	9	9	x 10^9	
Gold			x 10^{-1}	±5%
Silver			x 10^{-2}	±10%

(a) Resistor Color Code and Tolerance Chart

(b) 100 kΩ with a ±10% tolerance



(c) 6.8 kΩ with a ±10% tolerance

Fig. 5. Resistor Value (Ohm) Color Code and Tolerance Chart with Examples

After reviewing the chart, students identify and calculate resistance values using only the color code system, measure the same resistors with a digital multimeter, and compare theoretical and measured values. Discrepancies are analyzed in terms of tolerance or instrument precision, emphasizing the importance of accuracy and verification in experimental science.

Fig. 6. Ohm's Law verification circuit on a breadboard

Next, current, the flow of electric charge, is introduced. Current is measured in amperes (A). Resistance, measured in ohms (Ω), restricts this flow and depends on material, geometry, and temperature. The three quantities – voltage, current, and resistance – are related by Ohm's Law [7]:

$$V = I \times R \quad (5)$$

This relationship can be visualized through a triangle diagram and validated experimentally. Students build a circuit on a breadboard using a Keysight DC power supply, a multimeter, and a 100Ω resistor (**Fig. 6**).

By varying the voltage and recording the current, they verify Ohm's Law and generate a linear V–I plot. The slope of the line gives the resistance:

$$R = \frac{\Delta V}{\Delta I} \quad (6)$$

They also calculate the percent error relative to the nominal resistance:

$$\%\text{Error} = \frac{|R_{measured} - R_{actual}|}{R_{actual}} \times 100\% \quad (7)$$

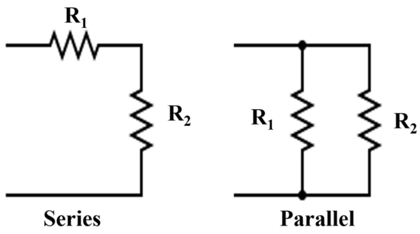
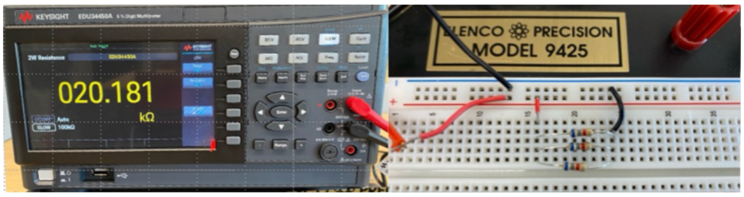


Fig. 7. Series and parallel resistor networks

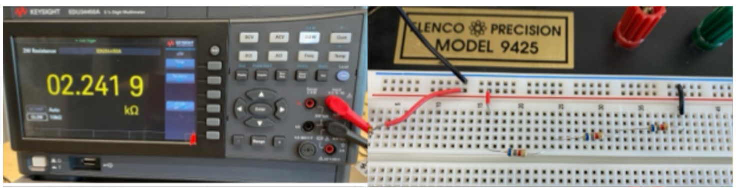
Once students are comfortable with single-element circuits, they extend this understanding to series and parallel connections [8] (**Fig. 7**). The governing relationships are:

$$\text{Series: } V_{\text{total}} = V_1 + V_2 + V_3, \quad I = I_1 = I_2 = I_3, \quad R_{\text{total}} = R_1 + R_2 + R_3 \tag{8}$$

$$\text{Parallel: } V = V_1 = V_2 = V_3, \quad I_{\text{total}} = I_1 + I_2 + I_3, \quad \frac{1}{R_{\text{total}}} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} \tag{9}$$



(a)



(b)

Fig. 8. Configuration of resistors on a breadboard: (a) parallel connection and (b) series connection.

Students assemble both series and parallel circuits on a breadboard (**Fig. 8**), measure the equivalent resistance with a multimeter, and compare results with theoretical predictions. Error analysis is included to highlight real-world deviations.

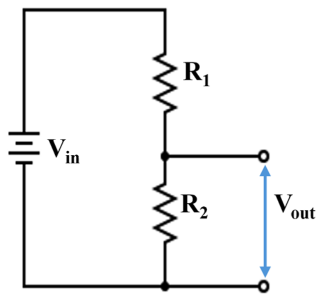


Fig. 9. Voltage divider circuit

A practical application is the voltage divider, a simple but widely used circuit (**Fig. 9**):

$$V_{\text{out}} = V_{\text{in}} \times \frac{R_2}{R_1+R_2} \tag{10}$$

By varying resistor values, students observe how the output voltage changes, thereby reinforcing the concept of proportionality in circuit design.

The final portion of this module introduces logic gates, the fundamental building blocks of digital electronics. Logic gates take one or more binary inputs and produce a single binary output, implementing Boolean operations such as NOT, AND, and OR. Gates are realized with transistors, semiconductor devices that act as both switches and amplifiers.

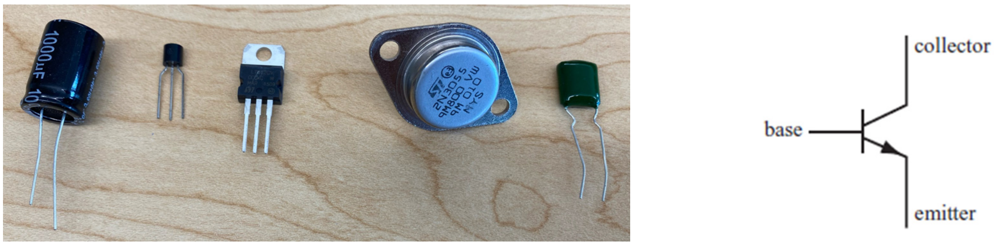


Fig. 10. Illustration of common transistor types along with the standard symbol for an NPN [9]

Before designing gates, students learn the basics of transistors [10, 11]. Doping silicon with group III elements (e.g., B, Al, Ga) produces p-type material (excess holes), while group V dopants (e.g., P, As, Sb) yield n-type material (excess electrons). Combining these produces PN junctions, which form the basis of NPN and PNP transistors (**Fig. 10**, **Fig. 11**).

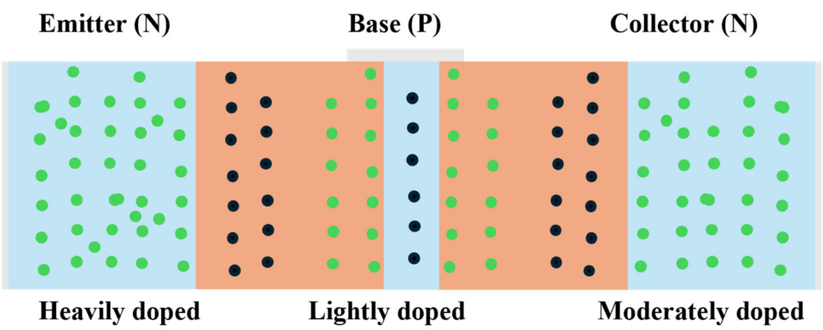


Fig. 11. Charge carriers in an NPN transistor

Table 2. Truth tables for NOT, AND, and OR gates

NOT Gate

Input	Output
A	Z
0	1
1	0

OR Gate

Input		Output
A	B	Z
0	0	0
0	1	1
1	0	1
1	1	1

AND Gate

Input		Output
A	B	Z
0	0	0
0	1	0
1	0	0
1	1	1

They then study truth tables for NOT, AND, and OR gates (**Table 2**) and build circuits to verify functionality. For instance, a NOT gate constructed with an NPN transistor (**Fig. 12**) demonstrates inversion: when the input is low (A = 0), the output LED glows (Z = 1); when the input is high (A = 1), the transistor conducts and the LED turns off (Z = 0), consistent with the truth table.

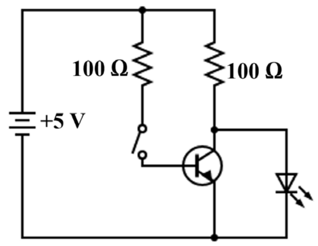


Fig. 12. NOT gate constructed with an NPN transistor

This exercise demonstrates how billions of such transistors can be combined into integrated circuits, the foundation of modern digital computation. By drawing parallels to quantum logic gates, the module helps students appreciate how classical electronics underpin the control and measurement systems necessary for quantum experiments.

3.3 Vacuum Technology

A vacuum system is indispensable in quantum research because it provides the clean, isolated environment necessary to preserve fragile quantum states. At atmospheric pressure, constant atomic and molecular collisions disturb quantum effects. By evacuating these particles, a vacuum enables stable, high-precision measurements, reduces contamination, and facilitates cryogenic operations.

The concept begins with pressure, defined as force per unit area:

$$P = \frac{F}{A} \tag{11}$$

where P is expressed in Pascals (Pa), the SI unit of pressure. Students compare SI units with commonly used ones – atmospheres (*atm*), *torr*, and *psi* – and practice conversions. For example:

$$1\ Pa = 0.00015\ psi = 0.00001\ atm$$

They also calculate atmospheric pressure at different altitudes using the exponential relation:

$$P(h) = P_0 \cdot e^{-h/8005} \tag{12}$$

where $P_0 = 1$ atm is the sea-level pressure and h is the altitude in meters. As an exercise, students evaluate whether the pressure on Mount Everest (~0.33 *atm*) qualifies as a vacuum. This activity connects the abstract definition of pressure to a real-world scenario that students can readily visualize.

Fig. 13. Edwards roughing pump, Pfeiffer turbomolecular pump, and Kurt J. Lesker convection gauge

To establish low pressures in practice, students are introduced to roughing pumps (operating in the 10^{-3} mbar range), turbomolecular pumps (10^{-9} mbar range), and pressure gauges used for monitoring. **Fig. 13** shows typical equipment, including an Edwards roughing pump, a Pfeiffer turbomolecular pump, and a Kurt J. Lesker convection gauge.

Fig. 14. KF and CF flanges with components required for assembly

Students also learn the differences between KF flanges, noted for their ease of replacement and lower cost, and CF flanges, which are used for ultra-high vacuum environments requiring long-term sealing (**Fig. 14**).

Hands-on training is central to this module. Students:

- Assemble KF and CF flanges using O-rings and copper gaskets.
- Connect a rotary pump to a custom-made chamber.
- Measure chamber pressure before and after pump-down.
- Perform a leak check with acetone or a residual gas analyzer (RGA).

Fig. 15. Top: Pressure gauge controller and custom-made vacuum chamber. Bottom: Pressure before and after pumping down the chamber

Through these activities, students not only gain experience handling real laboratory hardware but also develop an appreciation for the practical challenges of creating and maintaining vacuum conditions.

Fig. 15 illustrates a pressure gauge controller connected to a custom-made chamber, with recorded measurements before and after pumping down the system. By the end of this module, students will understand the fundamentals of vacuum systems, develop practical skills in assembling and operating pumps and flanges, and recognize the critical role of vacuum technology in quantum research and related industries.

3.4 Cryogenic Measurements

Quantum phenomena such as superconductivity, superfluidity, and qubit coherence are only accessible at cryogenic temperatures. Cooling reduces thermal noise, allowing fragile states to emerge. Training in cryogenic techniques, therefore, prepares students for work with dilution refrigerators, cryostats, and other instruments central to quantum research.

This module begins with a review of material classifications: conductors, insulators, semiconductors, resistors, and superconductors. Students connect these categories to everyday examples such as copper wire (conductor), silicon (semiconductor), and plastic (insulator). A superconductor is defined as a material with zero resistance below its critical temperature (T_c). For example, liquid helium cooling revealed superconductivity in mercury at 4 K [12]. The Bardeen–Cooper–Schrieffer (BCS) theory [13] later provided a microscopic explanation of this effect [14]. Another defining property is the Meissner effect, the expulsion of magnetic fields from the superconducting interior.

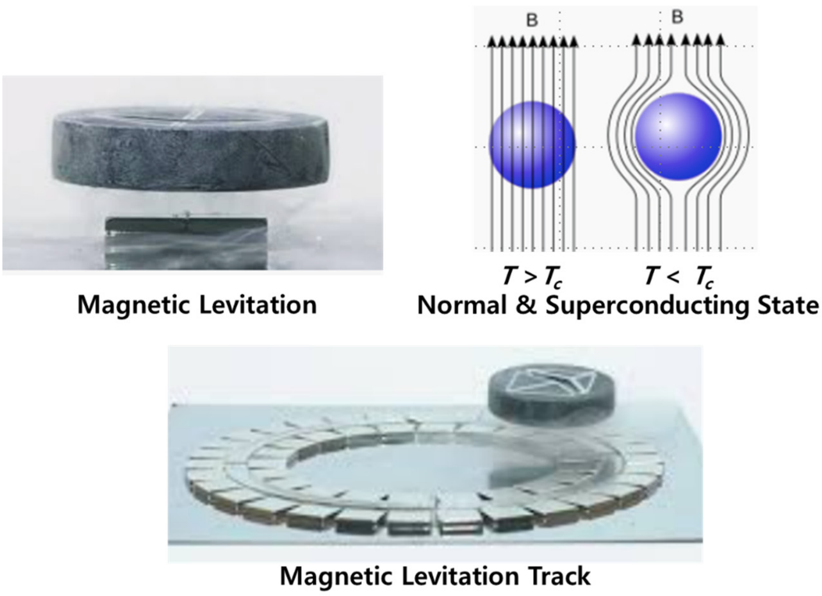


Fig. 16. Magnetic levitation of a YBCO superconductor illustrating the Meissner effect and flux pinning

The first hands-on exercise demonstrates magnetic levitation using Yttrium Barium Copper Oxide (YBCO). Since YBCO has a critical temperature of approximately 92 K [15], immersion in liquid nitrogen (77 K) renders it superconducting. When placed above a permanent magnet, the sample levitates due to flux expulsion and flux pinning, providing a vivid illustration of the Meissner effect [16]. Students also experiment with levitation tracks for dynamic demonstrations (**Fig. 16**). Safety protocols – including protective goggles and cryogenic gloves – are strictly enforced throughout the exercise.

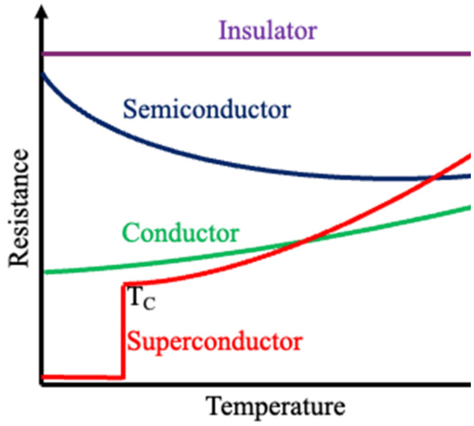
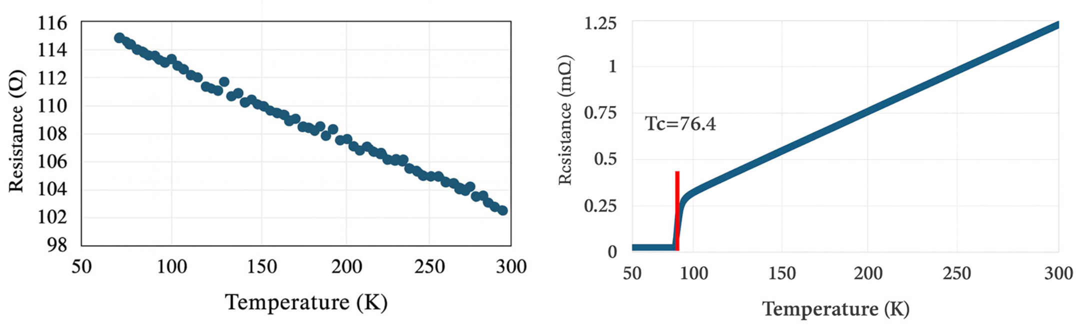


Fig. 17. Resistivity of different material types as a function of temperature, illustrating their fundamental differences.

Fig. 18. Cryocooler setup and electrical wiring configuration for the resistivity experiment, showing measurements for both a normal (non-superconducting) resistor and a superconducting YBCO sample.



(a) Standard 100 Ω resistor (b) YBCO sample with Tc = 76.4 K

Fig. 19. Resistivity versus temperature measurements after cooling. The left plot shows the behavior of a standard 100 Ω resistor, while the right plot displays the YBCO sample exhibiting a superconducting transition with a critical temperature of Tc = 76.4 K.

The experiment, illustrated in **Fig. 17**, shows the resistivity of different material types as a function of temperature, highlighting their fundamental differences. Students begin by analyzing how conductors, semiconductors, insulators, and superconductors behave under varying thermal conditions, identifying characteristic trends for each class of material. The second experiment involves measuring the resistance of materials as a function of temperature using a cryocooler setup, as shown in **Fig. 18**. In this hands-on activity, students investigate and compare the electrical behavior of different materials by recording how their resistance changes as the samples are cooled to cryogenic temperatures. The resulting plots highlight distinct behaviors: a 100 Ω resistor exhibits a gradual increase in resistance with decreasing temperature (**Fig. 19(a)**), while a superconducting material displays a sharp transition—its resistance abruptly drops to nearly zero below the critical temperature (Tc = 76.4 K)—as demonstrated for niobium (YBCO) in **Fig. 19(b)**.Together, these experiments help students visualize the fundamental differences among material types and identify the defining feature of superconductivity. By coupling conceptual understanding with direct experimental observation, students are introduced to both the physics and laboratory techniques of cryogenic science. Through phenomena such as zero resistance and magnetic levitation, they gain insight into why cryogenic systems are essential for quantum research, where maintaining extremely low temperatures is critical to achieving and sustaining quantum coherence.

3.5 Microwave and Radiofrequency Systems

Microwave (MW) and radiofrequency (RF) systems function as the nervous system of quantum devices. Superconducting qubits, for instance, rely on precisely shaped microwave pulses to initialize, manipulate, and measure quantum states. By working directly with MW and RF components, students gain hands-on familiarity with the same tools used in state-of-the-art laboratories for quantum information processing. The module begins with a review of signal fundamentals:

- Frequency (f) – cycles per second (Hz).
- Wavelength (λ) – distance between repeating wave features.
- Amplitude (A) – wave strength or intensity.
- Phase (ϕ) – relative timing or offset between signals.

Students use a Keysight waveform generator and a digital oscilloscope to generate and visualize sinusoidal signals, observing how each parameter affects the waveforms. Students then explore the role of attenuators, which reduce signal power. Attenuation in decibels is:

$$\text{Attenuation (dB)} = 20\log\left(\frac{V_{\text{in}}}{V_{\text{out}}}\right) \tag{13}$$

where V_{in} and V_{out} are input and output voltages, respectively.

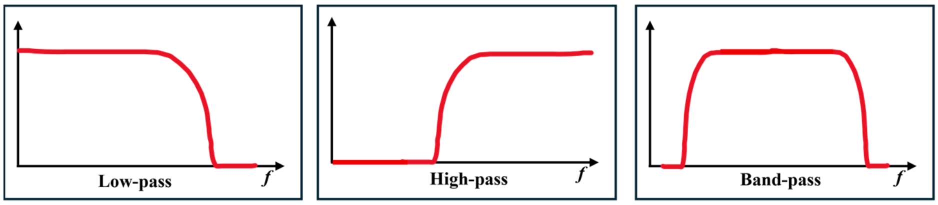


Fig. 20. Frequency response of low-pass, high-pass, and band-pass filters

Using the oscilloscope, students measure both quantities, calculate the attenuation, and confirm their agreement with the nominal values. This exercise is extended with a Vector Network Analyzer (VNA, Keysight P9374B), which provides frequency-domain measurements of component response. **Fig. 20** and **Fig. 21** illustrate the frequency response, showing how low-pass, high-pass, and band-pass filters are tested with the VNA to determine cut-off frequencies and transmission behavior.

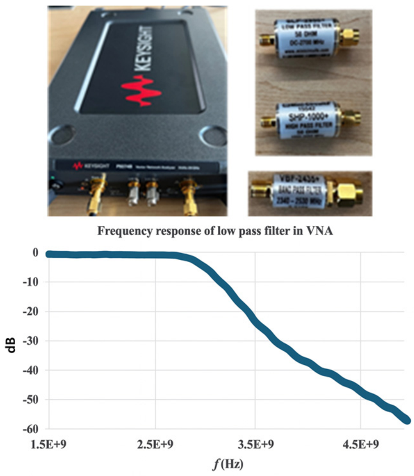


Fig. 21. VNA device, filters from Minicircuit, and VNA output after connecting a low-pass filter

Beyond attenuators and filters, students are introduced to additional RF components such as amplifiers and splitters, broadening their understanding of signal chains. Through these activities, they see how classical RF engineering principles are applied to shape and control the signals used to manipulate qubits.

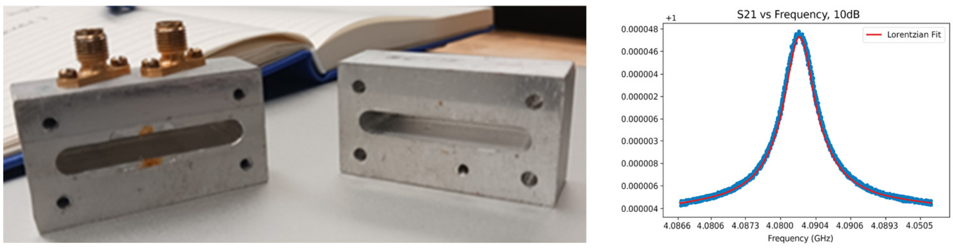


Fig. 22. Rectangular 3D cavity resonator and plot of signal intensity vs frequency at 10 dB

To connect RF theory directly with quantum hardware, students investigate a rectangular 3D microwave cavity resonator (**Fig. 22**). Such resonators are central to qubit readout and cavity quantum electrodynamics (cQED). A standard TE_{101} rectangular cavity with dimensions $a = 2.6924$ cm, $b = 0.762$ cm, and $d = 2.794$ cm is introduced. The resonant frequency is given by:

$$f_{mnl} = \frac{c}{2}\sqrt{\left(\frac{m}{a}\right)^2 + \left(\frac{n}{b}\right)^2 + \left(\frac{l}{d}\right)^2} \tag{14}$$

where m, n, and l are mode indices and c is the speed of light. For the TE_{101} mode, $m = 1$, $n = 0$, $l = 1$.

Students measure the transmission spectrum of the cavity using the VNA. At room temperature, the resonance peak appears broad with a relatively low quality factor (Q). When the cavity is cooled to 77 K with liquid nitrogen, the increased conductivity of the cavity walls reduces resistive losses, narrowing the resonance and increasing Q. The quality factor is calculated from the resonance linewidth:

$$Q = \frac{f_{\text{res}}}{\Delta f} \tag{15}$$

where f_{res} is the resonance frequency, and Δf is the full width at half maximum (FWHM).

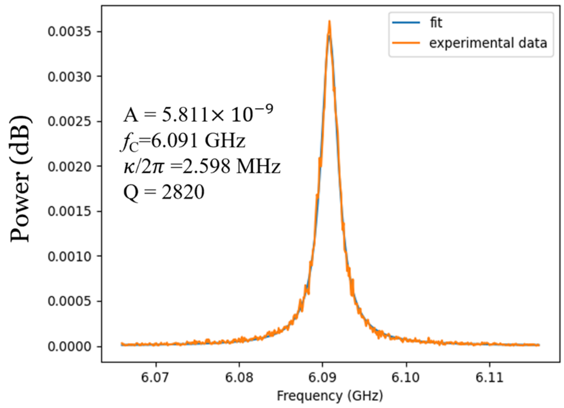


Fig. 23. Frequency vs. power response of the 3D cavity under liquid nitrogen cooling, with fitted curve providing the measured quality factor

Fig. 23 shows an example resonance curve, where students extract Q and compare results between room temperature and cryogenic operation. Through this experiment, students see how cryogenic environments significantly improve cavity performance, a principle that underpins the use of superconducting resonators in quantum information systems. At millikelvin temperatures, such cavities achieve exceptionally high-quality factors, enabling long photon lifetimes and coherent qubit–cavity interactions.

3.6 Quantum Simulations and Computations

This final module integrates knowledge from all previous modules into practical exercises in quantum computing. Students transition from foundational skills in programming, electronics, vacuum systems, cryogenics, and microwave engineering to designing, simulating, and executing quantum algorithms. The focus is on building a conceptual and practical bridge between classical hardware control and quantum computation.

The module begins with the concept of a qubit, the fundamental unit of quantum information. Unlike classical bits that exist in definite states 0 or 1, qubits can exist in superpositions of both states:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \text{ with } |\alpha|^2 + |\beta|^2 = 1 \tag{16}$$

where α and β are complex probability amplitudes. Measurement collapses the qubit into one of the two basis states, with probabilities determined by the amplitudes.

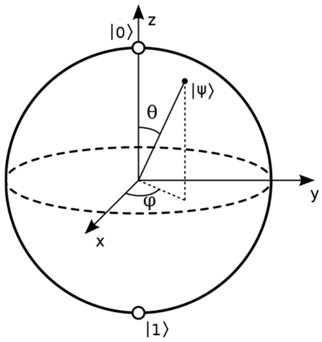


Fig. 24. Bloch sphere representation of a single qubit state

Students then study single-qubit gates, which correspond to rotations on the Bloch sphere as shown in **Fig. 24**. The Pauli-X gate flips the qubit state, the Hadamard gate creates superpositions, and phase gates introduce controlled rotations:

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{17}$$

In addition to the single-qubit gates introduced above, the two-qubit CNOT (CX) gate is represented by the following 4×4 matrix:

$$CX = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{18}$$

This matrix completes the mathematical description of the Bell-state circuit implemented in this module. These gates are implemented in simulation first, allowing students to visualize state vector evolution. To move beyond simulation, students employ Qiskit [17], IBM's open-source quantum computing framework. Using Python as the programming interface, students design circuits, run them on simulators, and finally execute them on real quantum hardware via IBM Quantum's cloud platform [18].

```python
from qiskit import QuantumCircuit, Aer, execute
from qiskit.visualization import plot_histogram

# Create a quantum circuit with 2 qubits and 2 classical bits
qc = QuantumCircuit(2, 2)

# Step 1: Apply Hadamard gate on qubit 0
qc.h(0)

# Step 2: Apply CNOT gate with qubit 0 as control and qubit 1 as target
qc.cx(0, 1)

# Step 3: Measure both qubits
qc.measure([0, 1], [0, 1])

# Simulate the circuit
simulator = Aer.get_backend('qasm_simulator')
result = execute(qc, simulator, shots=1024).result()
counts = result.get_counts(qc)

# Print results
print(counts)

# Plot histogram
plot_histogram(counts)
```

Fig. 25. Qiskit implementation of a Bell state circuit

A sample example is shown in **Fig. 25**, where a Bell state is generated by applying a Hadamard gate on the first qubit, followed by a CNOT gate: 1) Initializing qubits in the $|0\rangle$ state, 2) Applying Hadamard and CNOT gates to generate a Bell pair, 3) Executing the circuit on the Qiskit Aer simulator to verify ideal behavior, and 4) Running the same circuit on an IBM Quantum device, comparing outcomes with the simulator.

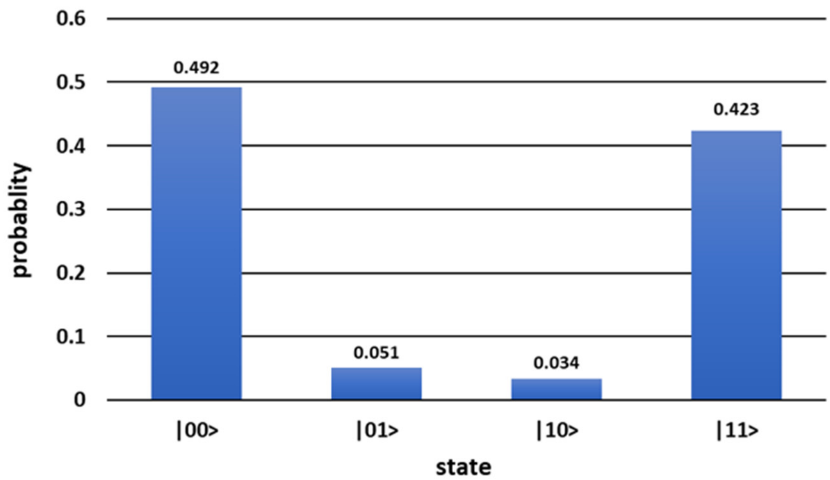


Fig. 26. Histogram of Bell state measurement outcomes showing deviations due to noise

The real-hardware results in **Fig. 26** were obtained using the IBM Quantum backend ibmq_osaka, a 127-qubit Falcon r5.11 processor. The experiment was executed with 1024 shots, and the calibration snapshot at the time reported relaxation times of $T_1 \approx 70–90$ μs, dephasing times of $T_2 \approx 60–80$ μs, single-qubit gate errors on the order of 10^{-4}, CNOT gate errors around 1–1.4%, and readout assignment errors of 2–4%. These hardware characteristics directly contributed to the deviations observed in the experimental results.

Compared with the ideal Bell-state distribution obtained in simulation, the hardware results show small but noticeable populations in the $|01\rangle$ and $|10\rangle$ states and an imbalance between $|00\rangle$ and $|11\rangle$. These deviations arise from CNOT gate infidelity, dephasing during circuit execution, and readout assignment errors, all of which are typical for NISQ devices. Discussing these effects helps students understand why real-hardware outcomes differ from ideal theoretical predictions.

These discrepancies also provide an ideal opportunity to introduce students to practical error-mitigation techniques. In this module, they examine how relaxation, dephasing, gate errors, and crosstalk affect circuit behavior and learn basic mitigation methods such as measurement-error mitigation, zero-noise extrapolation, randomized compiling, and transpiler optimization. This discussion helps students interpret noise-limited quantum computations and strengthens the hardware-aware focus of the curriculum.

By the conclusion of this module, students not only gain experience programming quantum computers but also develop the critical insight to interpret results in the context of real hardware limitations. This holistic approach prepares them to transition from classroom learning to active participation in experimental quantum research.

4. Educational Impact

The six modules outlined above introduce the basics of quantum science through detailed examples and demonstrations, along with the fundamental theory. This approach helps students become familiar with experiments, analysis, and critical thinking skills essential for research. The modules start with basic concepts and advance to more complex topics. As a result, students with no prior knowledge can understand and perform the experiments. By the end of the program, students will be able to explain why these experiments are necessary and important in quantum science research. This knowledge links theory with practical application, preparing them to work with the complex hardware supporting quantum technologies.

A simplified version of these modules was successfully implemented for high school students during the summer workshop at WSSU. Although students were initially unfamiliar with the concepts and experiments, they quickly understood the theory and felt comfortable performing the experiments. Feedback, shown in **Fig. 27**, indicated that they were enthusiastic about participating and appreciated the opportunity to explore quantum science. Students rate the overall experience as mostly excellent and the quality of the instructional content as high. The workshop also helped students understand the subject matter, and the hands-on training was highly beneficial. All the survey results are summarized in **Fig. 27**.

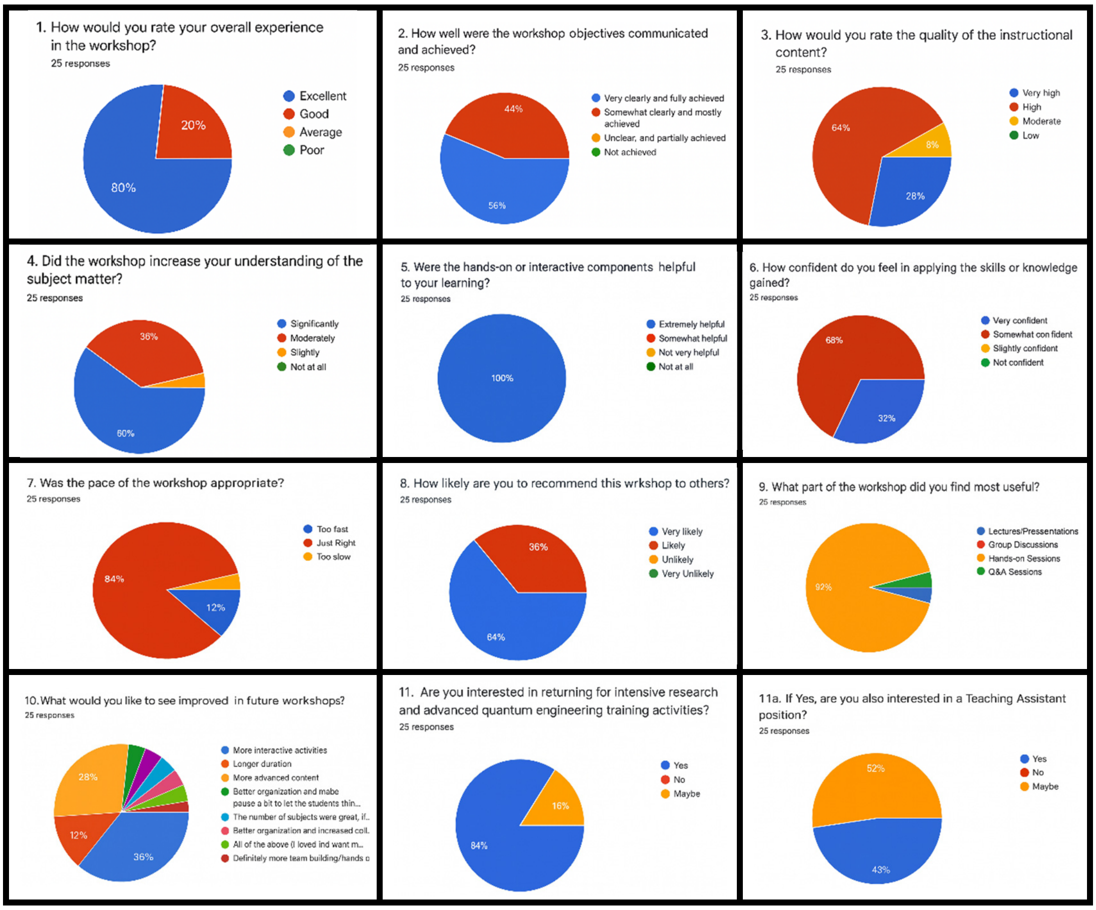


Fig. 27. Survey results collected from students after completing the summer workshop at WSSU

These experiments can be improved and expanded for future sessions and returning students, including undergraduates. For example, simple sample fabrication and characterization can be incorporated into the vacuum systems and cryogenic measurement modules. Students were asked what additional modules or experiments they want to see in future workshops. **Fig. 28** shows a bar chart summarizing survey responses (23 respondents) on desired additional modules. The chart shows strong interest in quantum simulation and computing (82.6%), quantum fabrication and materials (73.9%), cryogenic measurements at 4K (69.6%), qubit measurements and analysis (69.6%), microwave cavity resonator measurements (65.2%), and atomic force microscopy scanning (56.5%), with 14.3% finding all options interesting. These experiments or teaching materials can be incorporated for returning students as well as for undergraduates.

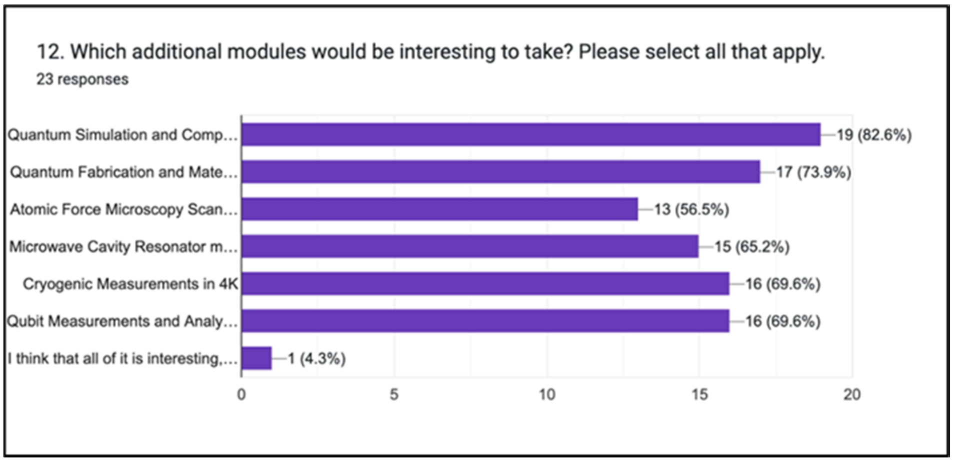


Fig. 28. Survey results collected from students on desired additional modules

As stated earlier, a simplified version was already implemented as a summer workshop. Since the modules start with the basic introduction, they are suitable for students with no prior knowledge or experience. However, as the modules progress, they delve into more complex theory and experiments. Nevertheless, these modules are designed to be expanded into much deeper theory, along with corresponding experiments. One example would be constructing gates such as AND, OR, NAND, etc., using multiple transistors. Such extensions can easily help turn this into a series of workshops. Keeping that in mind, a survey with different modules was conducted to see if students want additional modules for future workshops (**Fig. 28**). These additional teaching materials can be included for returning students. However, all these materials can also be seamlessly adapted into a single undergraduate course with detailed theory and experiments. The authors are also working on converting this into an after-hours online course for high school students or first-year undergraduates. Except for in-lab experiments, the other materials are designed for easy online access. In-lab experiments can also be accommodated through simulation experiments.

Through these modules, students gained practical understanding of fundamental concepts across electronics, vacuum systems, cryogenics, and RF technologies. In the electronics module, they worked with resistors, LEDs, and basic circuits, verifying series/parallel connections and using voltage dividers to relate theory to real measurements. The vacuum module taught them how pumps, gauges, and leak-detection methods function, along with hands-on skills such as operating flanges and fittings. Cryogenic measurements introduced superconducting materials, magnetic levitation, and safety procedures at low temperatures.

Finally, the MW/RF module familiarized students with signal basics and common RF components used for filtering, tuning, and device characterization.

Along with these foundational modules, students also progressed toward applied projects closely connected to active research. As part of our program, undergraduates developed secure quantum communication protocols, including a Quantum Authentication Protocol for Secure Superdense Coding [19] and a Quantum Digital Signature protocol [20]. Through these projects, students not only learned how entanglement and operator application enable secure communication but also gained an understanding of the vulnerabilities of quantum channels and how authentication layers can help prevent attacks.

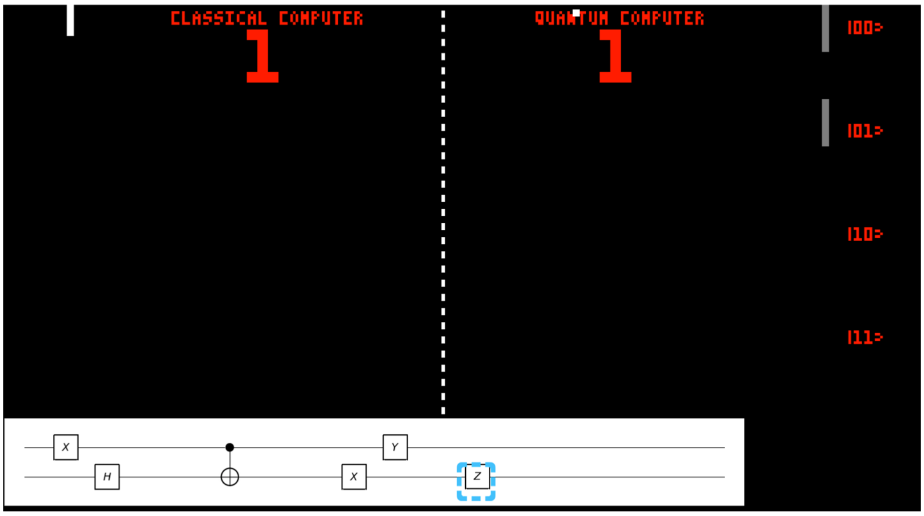


Fig. 29. Quantum Mini Tennis game, where players select 2-qubit system size

Students also interacted with quantum gates through the Quantum Mini Tennis game [21], an educational tool designed to illustrate the contrast between classical and quantum behavior. As shown in **Fig. 29**, the interface displays a classical score on the left and quantum-state outcomes on the right, including the probability amplitudes of the basis states $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$. The gate sequence at the bottom allows students to apply X, H, Y, Z, and CX operations, enabling them to observe in real time how these gates transform the quantum state. This visualization helps students connect gate operations to changes in probability distributions and better understand the differences between classical logic and quantum state evolution. During actual gameplay, students select quantum gates instead of directly controlling a paddle. Each chosen gate updates the qubit state, and the resulting state determines the paddle's movement and the ball's trajectory in the quantum court. In contrast, the classical side follows predetermined, deterministic rules. By experimenting with different gate sequences, students experience how quantum operations—superposition, phase shifts, and entanglement—affect the game mechanics, scoring, and overall dynamics. This interactive structure reinforces conceptual understanding while allowing learners to 'feel' the behavioral differences between classical and quantum computation through gameplay.

These extensions highlight the educational impact of our program, as students progressed from basic labs to implementing protocols and simulations that bridge theory with real-world applications. This hardware-aware approach shows how quantum education lowers entry barriers while empowering students to contribute to research and innovation in secure communication and quantum software.

All the modules are fundamentally connected to the fabrication and study of quantum devices. Python programming and quantum simulation are crucial for simulating a wide range of devices. Simultaneously, electronics and circuits, vacuum technology, cryogenic measurements, and microwave and radiofrequency systems are vital for the fabrication and characterization of real quantum devices. Students were able to identify the connections between these different modules at the end.

Furthermore, the effectiveness of our hardware-aware curriculum is supported by established educational theories. Its hands-on, experiment-driven structure aligns with Kolb's Experiential Learning Theory and key constructivist principles, as students actively build knowledge through direct engagement with hardware, simulations, and collaborative problem solving [22–24]. These frameworks together provide a strong pedagogical basis for the educational impact demonstrated in our program.

5. Conclusion

This paper presented a six-module, hardware-aware curriculum bridging quantum theory and practical hardware realities. Covering Python programming, electronics, vacuum systems, cryogenics, microwave engineering, and quantum simulations, it provides a structured path for students from diverse backgrounds to engage with quantum technologies. Implementation at WSSU showed that hands-on learning lowers entry barriers while enhancing confidence and teamwork. Through activities such as circuit building, vacuum assembly, magnetic levitation, and Qiskit-based programming, students connected theory to practice. Research-driven projects—including secure quantum communication protocols and the Quantum Mini Tennis game—fostered creativity and innovation. This framework offers a scalable model for integrating hardware-aware quantum education into undergraduate curricula, preparing the next generation for the quantum workforce.

Although the present curriculum does not yet include a standalone module on quantum error mitigation or correction, several foundational mitigation concepts—such as interpreting calibration data, understanding noise sources, and applying basic error-reduction techniques—are introduced within the Quantum Simulations and Computations module. In future iterations, we plan to expand these topics into a dedicated seventh module focusing specifically on quantum error mitigation and introductory quantum error correction.

Acknowledgements

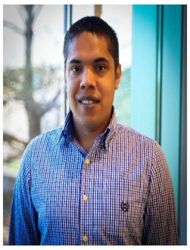
The authors acknowledge support from the U.S. National Science Foundation (Grant No. 2329017) and the U.S. Department of Energy (Grant No. DE-SC0023595). The authors also acknowledge Dr. Kasra Sardashti at the Laboratory for Physical Sciences (LPS) /University of Maryland (UMD) for their support in microwave and cavity measurements.

References

[1] N. Sobrino, U. Aseginolaza, J. Jornet-Somoza, and J. Borge, "Refining Noise Mitigation in NISQ Hardware Through Qubit Error Probability," *arXiv preprint arXiv:2503.10204*, Mar. 2025. Article (CrossRef Link)

[2] Y. Zhu et al., "Hardware-aware Calibration Protocol for Quantum Computers," in *Proc. of the 52nd Annual International Symposium on Computer Architecture (ISCA '25)*, pp.241-256, 2025. Article (CrossRef Link)

[3] F. Orts, "Pedagogical Approach for Quantum Computing in Engineering Education," *Computer Applications in Engineering Education*, vol.33, no.2, Mar. 2025. Article (CrossRef Link)
[4] A. Virani, Devraj, A. Suresh, L. Zhang, and M. V. P. Rao, "Distinguishing Quantum Software Bugs from Hardware Noise: A Statistical Approach," *arXiv preprint arXiv:2507.20475*, Jul. 2025. Article (CrossRef Link)
[5] TE Connectivity, Resistor Color Codes: What Do the Color Bands Mean?. [Online]. Available: https://www.te.com/en/products/passive-components/resistors/intersection/resistor-color-codes.html
[6] G. S. Ohm, The Galvanic Circuit Investigated Mathematically, D. Van Nostrand Company, 1905.
[7] T. R. Kuphaldt, Lessons in Electric Circuits, Volume I – DC, Vol. Fifth Edition, Open Book Project, 2006. Article (CrossRef Link)
[8] E. Boysen and H. Kybett, Complete Electronics Self-Teaching Guide with Projects, John Wiley & Sons, 2012. Article (CrossRef Link)
[9] S. W. Amos, Principles of Transistor Circuits: Introduction to the Design of Amplifiers, Receivers, and Digital Circuits, Elsevier, 2013.
[10] K. G. Nichols, and E. V. Vernon, Transistor Physics, Springer Science & Business Media, 2013.
[11] H. K. Onnes, "The disappearance of the resistance of mercury," Comm. Phys. Lab. Univ. Leiden, vol.122b, 1911.
[12] J. Bardeen, L. N. Cooper, and J. R. Schrieffer, "Theory of Superconductivity," *Physical Review Journals Archive*, vol.108, no.5, pp.1175-1204, Dec. 1957. Article (CrossRef Link)
[13] W. Meissner and R. Ochsenfeld, "Ein neuer Effekt bei Eintritt der Supraleitfähigkeit," *Naturwissenschaften*, vol.21, no.44, pp.787-788, Nov. 1933. Article (CrossRef Link)
[14] A. A. A. Hussein, A. M. A. Hussein, and N. A. Hasan, "Study of the Properties of YBCO Superconductor Compound in Various Preparation Methods: A Short Review," *Journal of Applied Sciences and Nanotechnology*, vol.3, no.1, pp.65-79, 2023. Article (CrossRef Link)
[15] Quantum Levitation, Superconductor Experiment Kits. [Online]. Available: https://quantumlevitation.com/product/superconductivity-kit/
[16] H. Kim, Y. Jung, Y. W. Jo, M. K. Lee, J. W. Choi, Y. Kim, J. Kim, W.-G. Paeng, M. S. Kim, H. Jung, and Y. K. Kwon, "Temperature Measurement Techniques for RAON Cryomodule," *Applied Science and Convergence Technology*, vol.27, no.2, pp.30-34, 2018. Article (CrossRef Link)
[17] G. Aleksandrowicz, Open-source quantum software for large-scale execution, 2019. [Online]. Available: https://zenodo.org/records/2562111 (last accessed in May 2026)
[18] IBM Quantum Platform. [Online]. Available: https://quantum.cloud.ibm.com/ (last accessed in May 2026)
[19] T. Kim, J. Baek, and J. T. Yi, "Quantum Authentication Protocol for Secure Quantum Superdense Coding," in *Proc. of SoutheastCon 2024*, pp.863-864, 2024. Article (CrossRef Link)
[20] A. O. Hijaz, I. Yakubu, and J. Baek, "Integrating Secure Quantum Digital Signature into Quantum Communications," in *Proc. of International Conference on Grid, Cloud, and Cluster Computing; Quantum Technologies; and Modeling, Simulation and Visualization Methods (CSCE 2024)*, Communications in Computer and Information Science, vol.2257, pp.165-173, 2025. Article (CrossRef Link)
[21] M. E. Morgan, A. O. Hijaz, and J. Baek, "WSSU Quantum Mini Tennis: Learning Quantum Gates and State Vectors Through Gaming," in *Proc. of ADMI 2024 Conference*, Apr. 2024.
[22] A. Y. Kolb, and D. A. Kolb, "Experiential Learning Theory: A Dynamic, Holistic Approach to Management Learning, Education and Development," *The SAGE Handbook of Management Learning, Education and Development*, 2009. Article (CrossRef Link)
[23] S. Radović, H. G. K. Hummel, and M. Vermeulen, "The Challenge of Designing 'More' Experiential Learning in Higher Education Programs in the Field of Teacher Education: A Systematic Review Study," *International Journal of Lifelong Education*, vol.40, no.5-6, pp.545-560, 2021. Article (CrossRef Link)
[24] A. S. Palincsar, "Social constructivist perspectives on teaching and learning," *Annual Review of Psychology*, vol.49, no.1, pp.345-375, Feb. 1998. Article (CrossRef Link)



Shiva Dahal is a postdoctoral research associate in the Chemistry Department at Winston-Salem State University (WSSU), North Carolina, USA. He earned his Ph.D. in Physics from the University of Tennessee in 2024. His research focused on growing and characterizing thin films, including their structural and electronic properties, in ultra-high vacuum systems. As a postdoctoral researcher at WSSU, he developed training modules related to Quantum Information Science and Engineering (QISE) for undergraduate students under the supervision of Professor John T. Yi. He is also working to set up a new quantum laboratory at WSSU capable of conducting quantum research, equipped with a sputtering deposition chamber, a dilution refrigerator, and atomic force microscopy. Additionally, he is involved in nanofabricating superconducting islands on thin metal films to study proximity-induced superconductivity, which will aid in the development of practical quantum devices.

John T. Yi is a Professor of Chemistry at Winston-Salem State University (WSSU), North Carolina, USA. He received his Ph.D. in Physical Chemistry from the University of Pittsburgh in 2005 and conducted postdoctoral research at The Ohio State University and Stanford University before joining WSSU in 2006. Dr. Yi's research began with high-resolution molecular spectroscopy in the terahertz (THz) and ultraviolet (UV) regions, focusing on the structures and dynamics of biomolecules. Expanding his expertise, he has become a leader in quantum information science and engineering (QISE), with research spanning quantum devices, superconducting materials, and quantum photonics. His current work investigates coherent transduction of quantum states between optical and microwave photons, with applications in circuit quantum electrodynamics (cQED), quantum computing, and quantum communication. Dr. Yi has secured multiple federal grants to build advanced quantum laboratories and to train a diverse next-generation quantum workforce. His contributions bridge chemistry, spectroscopy, and quantum technology, establishing WSSU as a hub for interdisciplinary quantum research.

Jinsuk Baek is a Professor of Computer Science at Winston-Salem State University (WSSU), North Carolina, USA. He received his Ph.D. in Computer Science from the University of Houston in 2004 and joined WSSU the following year. Dr. Baek's research spans quantum computing, networking, security protocols, and 3D printing methodologies, with his work published in leading journals and conferences. Recently, he developed an innovative 3D food printer capable of producing edible chocolate, a breakthrough that has attracted significant attention in the field of food technology. He has been awarded multiple grants to advance research and education in quantum computing as well as AI-driven robotics for food technology. Dr. Baek's academic and industrial contributions are widely recognized, and he continues to pioneer new directions at the intersection of computing, quantum science, and emerging technologies.