Implementing The B92 Quantum Key Distribution Protocol

1. Overview

The **B92 Quantum Key Distribution (QKD)** protocol is a prepare-and-measure QKD scheme introduced by **Charles Bennett in 1992**, which uses **two non-orthogonal quantum states** to encode classical bits. This minimalist design makes it **easier to implement** while retaining the core quantum security features.

2. Motivation for Implementation

The B92 protocol provides a **resource-efficient alternative** for quantum-secure key exchange. Due to its **reduced complexity**, it is especially suitable for constrained environments and can be deployed using standard photonic platforms. Its use of **only two quantum states** makes it ideal for experimental setups and embedded quantum processors where gate count and circuit depth are limited.

Implementing B92 within Classiq's platform will:

- Provide users with a lightweight QKD protocol option.
- Expand Classiq’s QKD protocol support beyond four-state systems.
- Enable novel applications requiring simpler quantum hardware.

3. Protocol Workflow

Step 1: State Preparation

- Alice generates a random bit string.
- Bit 0 is encoded as quantum state $|\psi_0\rangle = |0\rangle$.
- Bit 1 is encoded as $|\psi_1\rangle = |+\rangle$, a non-orthogonal state relative to $|0\rangle$.

Step 2: Quantum Transmission

- Qubits are transmitted over a quantum channel from Alice to Bob.

Step 3: Measurement by Bob

- Bob attempts to determine the state using projective measurements in a basis that allows partial distinguishability.
- For example, Bob may measure in the orthogonal basis of the *other* state: if Alice sends $|0\rangle$, Bob measures in $\{|1\rangle, |-\rangle\}$.

- If Bob receives a conclusive result (i.e., detects orthogonal state), he records a bit. Otherwise, he discards it.

Step 4: Key Sifting

- Alice and Bob publicly compare indices of detected bits.
- Bits where Bob got no conclusive result are discarded.
- The remaining bits form the **sifted key**.

Step 5: Post-Processing

- **Error correction** removes discrepancies between Alice and Bob's keys.
- **Privacy amplification** reduces any partial knowledge an eavesdropper (Eve) might have gained.

4. Quantum Circuit Model for B92

A Classiq-compatible implementation would involve:

Alice's Circuit:

- Random bit generator.
- State preparation:
    - Bit 0: Apply identity gate ($|0\rangle$).
    - Bit 1: Apply Hadamard gate to prepare $|+\rangle$.

Bob's Circuit:

- Measurement setup with basis selection aligned to B92 requirements:
    - Use of Hadamard and Pauli-X gates to define rotated measurement basis.
- Classical post-selection logic to discard inconclusive detections.

5. Simulation and Testing Guidelines

A full simulation should include:

- Varying photon loss and detector efficiency.
- Measuring **Quantum Bit Error Rate (QBER)**.
- Recording the **key generation rate** under different loss models.
- Evaluating security against intercept-resend attacks (e.g., using a simulated Eve).

6. Deliverables

- Add **B92 protocol support** to its quantum algorithm library.
- Provide an **abstracted circuit builder** for B92 (like currently supported BB84/Bell circuits).
- Support **parameterized simulations** for B92:
    - QBER tracking
    - Sifted vs. raw key rates
- Integrate **visual debugging** for state discrimination and loss impact.

7. Conclusion

The **B92 QKD protocol** offers a streamlined approach to quantum cryptography using only **two non-orthogonal states**, making it ideal for low-complexity, scalable quantum systems. Adding B92 support to the Classiq platform will extend its usability in experimental and applied quantum security scenarios.