information

IMPACT FACTOR 2.9

CITESCORE 6.5

Article

New Circuits for Simultaneously Initiating Two Different Quantum Superpositions

Artyom M. Grigoryan and Alexis A. Gomez

MDPI

https://doi.org/10.3390/info16121043

information

MDPI

Article

New Circuits for Simultaneously Initiating Two Different Quantum Superpositions

Artyom M. Grigoryan * and Alexis A. Gomez

Department of Electrical and Computer Engineering, The University of Texas at San Antonio, San Antonio, TX 78249, USA; alexis.gomez@my.utsa.edu
* Correspondence: artyom.grigoryan@utsa.edu

Abstract

This article presents, for the first time, a new approach to building quantum circuits for the initialization of two multi-qubit superpositions, namely, two different superpositions in one circuit, not in two separate circuits. For this, we introduce the concept of the discrete two signal-induced heap transformation (D2siHT). This transformation is generated by two signals, or vectors, which we call generators. The quantum analogue of the D2siHT is described. It allows us to build a quantum circuit to transform two superpositions $|\boldsymbol{x}\rangle$ and $|\boldsymbol{y}\rangle$ into the first conventual basis states $|000\ldots0\rangle$ and $|010\ldots0\rangle$, respectively. Therefore, we can build a single quantum circuit to initiate two multi-qubit superpositions $|\boldsymbol{x}\rangle$ and $|\boldsymbol{y}\rangle$ from the basis states $|000\ldots0\rangle$ and $|010\ldots0\rangle$, respectively. Examples with quantum circuits for the preparation and transformation of two 2- and 3-qubit superpositions are described in detail. The results of circuit simulation using Qiskit are also presented. The main characteristic of the D2siHT is its path of processing the data of two generators and input qubits. We consider different paths to effectively compute the D2siHT. Such paths can reduce, for instance, the depth of the resulting quantum circuits, which can lead to a reduction in execution times and susceptibility to decoherence and noise. Multi-qubit superpositions are considered with real amplitudes, but the presented approach can be extended to initiate two such superpositions with complex amplitudes, as well.

Keywords: qubit state preparation; quantum signal-induced heap transformation; Givens rotations

Academic Editor: Heming Jia

Received: 8 October 2025
Revised: 4 November 2025
Accepted: 28 November 2025
Published: 30 November 2025

Citation: Grigoryan, A.M.; Gomez, A.A. New Circuits for Simultaneously Initiating Two Different Quantum Superpositions. *Information* **2025**, *16*, 1043. https://doi.org/10.3390/info16121043

1. Introduction

In quantum computation, the main objects are qubits and the operations performed on them, which can only be unitary. Qubits of number r are mathematically represented as a linear superposition of 2^r possible basis states with amplitudes describing the probabilities of being in one of these states. The preparation of such superpositions representing different data, including signals and images, is an especially important task, and it is the first task in quantum computing [1–11]. The traditional approach to initiating a multi-qubit superposition $|\boldsymbol{x}\rangle$ starts with composing a unitary transformation of $|\boldsymbol{x}\rangle$ into the conventual basis state, for instance, the first one, as $U_x : |\boldsymbol{x}\rangle \rightarrow |00\ldots0\rangle$. Then, a quantum circuit is built to calculate the inverse transform, $U_x^{-1} : |00\ldots0\rangle \rightarrow |\boldsymbol{x}\rangle$. And because the operation is unitary, the inverse operator U_x^{-1} is easy to calculate. Based on this approach, another important task of transferring one multi-qubit superposition to another, $|\boldsymbol{x}\rangle \rightarrow |\boldsymbol{y}\rangle$, can be accomplished as well. This operation is fulfilled by two transformations to the same

conventual basis state, $U_x : |\boldsymbol{x}\rangle \rightarrow |00\ldots0\rangle$ and $U_y : |\boldsymbol{y}\rangle \rightarrow |00\ldots0\rangle$. This results in the traditional two-stage approach with the transformation $T = U_y^{-1}U_x : |\boldsymbol{x}\rangle \rightarrow |00\ldots0\rangle \rightarrow |\boldsymbol{y}\rangle$, which is known as the state-to-state preparation task. To initiate quantum superpositions, different methods can be used. We mention the Householder transformations [12–14] and Given rotations [15,16]. Methods with Givens rotations are simple, and many quantum circuits have been developed [17–19]. These circuits contain many permutations, namely, the CNOT and controlled CNOT operations. Therefore, much attention has been paid to minimizing the number of such gates. It was found that the number of elementary rotation gates is around $2(2^r - 1)$, and that of CNOTs is $2(2^r - r - 1)$ [20–22]. Such a large number of CNOTs is understandable because quantum circuits require all gates to operate on adjacent, or nearest-neighbour, bit planes (BPs). Permutations with Gray code are used for this purpose [22,23]. Recently, it was shown that a multi-qubit superposition $|\boldsymbol{x}\rangle$ can be obtained from another multi-qubit superposition $|\boldsymbol{y}\rangle$ in only one step. Namely, the $|\boldsymbol{x}\rangle \rightarrow |\boldsymbol{y}\rangle$ transformation can be accomplished without transferring the superpositions to the basis state $|00\ldots0\rangle$ [24]. This effective method is based on the discrete signal-induced heap transform (DsiHT) and uses only $(2^r - 1)$ rotation gates (in the real case of amplitudes) without any permutation [25].

In this work, we analyze the method of composing a unitary transformation which is composed of 3D rotations. Namely, we introduce the discrete two signal-induced heap transform (D2siHT), which is generated by two generators. These unitary transformations can be used to initiate two different multi-qubit superpositions with real amplitudes. The quantum analogue of the N-point D2siHT is the r-qubit quantum two signal-induced heap transform (Q2siHT). The key contributions of this work are the following:

1. We develop a unitary transformation generated by two vectors, or two quantum superpositions, by using 3D rotations.
2. We propose quantum circuits, each of which allows us to calculate two different multi-qubit superpositions.
3. We present examples with different paths for initiating two 2- and 3-qubit superpositions. These circuits are constructed, tested, and validated using the Qiskit framework.

The rest of this paper is organized as follows. The DsiHT with one generator is described in Section 2. In Section 3, the new concept of the D2siHT is presented, and examples are described in detail. In Section 4, we discuss the importance of the path in the initialization of the qubit superpositions with examples for 2- and 3-qubit superpositions. Finally, the presented circuits are analyzed using the Qiskit framework in Section 5.

2. The DsiHT with One Generator

The DsiHT is defined by a special selection of parameters that are initiated by vector-generators through so-called decision equations [26]. Let $f_0(\boldsymbol{z}, \vartheta_1, \ldots, \vartheta_m)$, …, $f_{N-1}(\boldsymbol{z}, \vartheta_1, \ldots, \vartheta_m)$ be parameterized functions of an N-dimensional vector variable $\boldsymbol{z} = (z_0, z_1, \ldots, z_{N-1})$, where $N > 1$, and $m \geq (N-1)$. Parameters $\vartheta_1, \ldots, \vartheta_m$ are chosen to adjust the transformation

$$T_{\vartheta_1,\ldots,\vartheta_m} : \boldsymbol{z} \rightarrow (f_0(\boldsymbol{z}, \vartheta_1, \ldots, \vartheta_m), \ldots, f_{N-1}(\boldsymbol{z}, \vartheta_1, \ldots, \vartheta_m)), \tag{1}$$

to make it, for instance, unique, separable, unitary, and fast. The transformation $T_{\vartheta_1,\ldots,\vartheta_m}$ is considered separable, which means that there exist such transformations $T_{\vartheta_1}, T_{\vartheta_2}, \ldots, T_{\vartheta_m}$ that

$$T_{\vartheta_1,\ldots,\vartheta_m} = T_{\vartheta_{i(m)}} \cdots T_{\vartheta_{i(2)}} T_{\vartheta_{i(1)}} \tag{2}$$

where $i(k)$ is a permutation of numbers $k = 1, 2, \ldots, m$.

Here, we limit ourselves with the case when each transformation $T_{\vartheta_{i(m)}}$ changes only two components of the vector $\boldsymbol{z}$. Thus, $T_{\vartheta_{i(k)}}$ is described as a transformation

$$T_{\vartheta_{i(k)}} : \boldsymbol{z} \to \left(z_0, \ldots, z_{k-1}, f_{k_1}(\boldsymbol{z}, \vartheta_k), z_{k_1+1}, \ldots, z_{k_2-1}, f_{k_2}(\boldsymbol{z}, \vartheta_k), z_{k_2+1}, \ldots, z_m\right), \tag{3}$$

where the pair of numbers (k_1, k_2) is uniquely defined by k, and $0 \le k_1 < k_2 \le m$. This means that these transformations are isomorphic to the two-dimensional transformations (see Figure 1)

$$T_{k_1,k_2}(\vartheta_k) : \left(z_{k_1}, z_{k_2}\right) \to \left(f_{k_1}\left(z_{k_1}, z_{k_2}, \vartheta_k\right), f_{k_2}\left(z_{k_1}, z_{k_2}, \vartheta_k\right)\right). \tag{4}$$

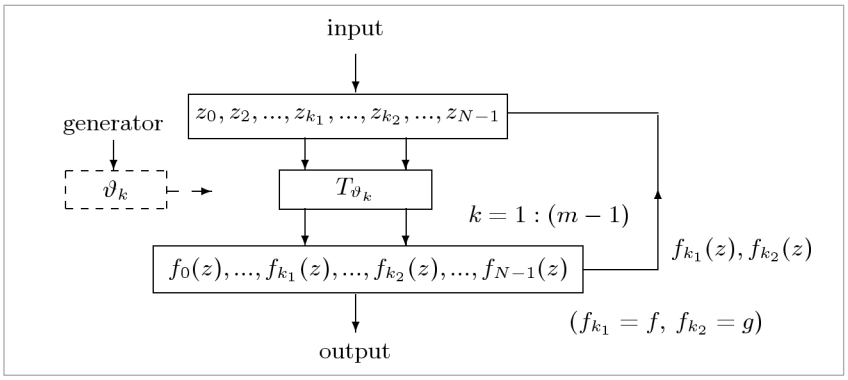


Figure 1. A block diagram of the DsiHT.

We add one more simplification by assuming that all the first functions $f_{k_1}\left(z_{k_1}, z_{k_2}, \vartheta_k\right)$ in this equation are equal to a function $f\left(z_{k_1}, z_{k_2}, \vartheta_k\right)$, and all functions $f_{k_2}\left(z_{k_1}, z_{k_2}, \vartheta_k\right)$ are equal to a function $g\left(z_{k_1}, z_{k_2}, \vartheta_k\right)$. Then, the problem of the construction of the N-D transformation $T = T_{\vartheta_1,\ldots,\vartheta_m}$ is reduced to defining unitary transformations

$$T_{k_1,k_2}(\vartheta_k) : \left(z_{k_1}, z_{k_2}\right) \to \left(f\left(z_{k_1}, z_{k_2}, \vartheta_k\right), g\left(z_{k_1}, z_{k_2}, \vartheta_k\right)\right). \tag{5}$$

And finally, we propose the following method of selecting parameters $\vartheta_k, k = 1 : m$. The selection of these parameters is based on specified vector-generators, the number of which is defined through a given system of decision equations, to achieve the uniqueness of parameters and desired properties of the transformation T. The number of decision equations defines the complexity of transformations.

First, we describe the case of two decision equations with one vector-generator $\boldsymbol{x}$. We assume that the functions f and g are linear,

$$f\left(z_{k_1}, z_{k_2}, \vartheta_k\right) = a_k z_{k_1} + b_k z_{k_2}, \tag{6}$$

$$g\left(z_{k_1}, z_{k_2}, \vartheta_k\right) = c_k z_{k_1} + d_k z_{k_2}, \tag{7}$$

where the coefficients $a_k,\ b_k,\ c_k$, and d_k are functions of ϑ_k. Thus, we have two functions $f(x, y, \vartheta)$ and $g(x, y, \vartheta)$. The parameter ϑ refers to the rotation parameter, such as the angle, and x and y the coordinates of the point (x, y) on the plane. These variables may have other meanings as well. It is assumed that, for a given set of parameters $a = a_1, a_2, \ldots,\ a_m$, the equation $g(x, y, \vartheta) = a$ has a unique solution with respect to ϑ, for each point (x, y) on the plane or its chosen subset. The system of equations

$$\begin{cases} f(x, y, \vartheta) = y_0 \\ g(x, y, \vartheta) = a \end{cases} \tag{8}$$

is called the system of decision equations. The value of ϑ is calculated from the second equation, which we call the angular equation. Then, the value of y_0 is calculated from the given input (x, y) and ϑ. It is also assumed that the two-point transformation

$$T_{\vartheta_k} : (z_{k_1}, z_{k_2}) \to \big(f(z_{k_1}, z_{k_2}, \vartheta_k), g(z_{k_1}, z_{k_2}, \vartheta_k)\big), \tag{9}$$

which is derived (or whose matrix is calculated) from the given decision equations by

$$T_{\vartheta_k} : (x_{k_1}, x_{k_2}) \to \big(f(x_{k_1}, x_{k_2}, \vartheta_k), a_k\big) \tag{10}$$

and which we call the basic transformation, is unitary. The transformations T_{ϑ_k} are the little building blocks from which we construct a unitary transformation.

As an example, we consider the decision equations with the elementary rotations

$$\begin{cases} f(x, y, \vartheta) = x\cos\vartheta - y\sin\vartheta, \\ g(x, y, \vartheta) = x\sin\vartheta + y\cos\vartheta. \end{cases} \tag{11}$$

The basis transformation is defined as the rotation of the point (x, y) to the horizontal $Y = a$,

$$T_\vartheta : (x, y) \to (x\cos\vartheta - y\sin\vartheta, a). \tag{12}$$

The rotation angle is calculated by

$$\vartheta = \operatorname{acos}\left(\frac{a}{\sqrt{x^2 + y^2}}\right) + \operatorname{atan}\left(\frac{x}{y}\right) \tag{13}$$

and $\vartheta = \operatorname{asin}(a/x)$, when $y = 0$. Here, the following condition is required: $a^2 \le x^2 + y^2$. Then, the basic transformations T_{ϑ_k} in Equation (9) are defined from the decision equations

$$\begin{cases} f(x_{k_1}, x_{k_2}, \vartheta_k) = x_{k_1}\cos\vartheta_k - x_{k_2}\sin\vartheta_k, \\ g(x_{k_1}, x_{k_2}, \vartheta_k) = a_k. \end{cases} \tag{14}$$

The special case when all parameters $a_k = 0$ leads to the N-point DsiHT, which operates on the generator as $H_N\boldsymbol{x} = (\pm\|\boldsymbol{x}\|, 0, \ldots, 0)\prime$. Here, the apostrophe denotes the transpose operation. The transform collects the generator's energy in one place or heap. The order of processing the components of the generator, which is called the path of the transformation, can be chosen in different ways. For instance, the following path can be considered

$$T_{\vartheta_1} : (x_0, x_1) \to \left(\widetilde{x}_0, 0\right), T_{\vartheta_2} : \left(\widetilde{x}_0, x_2\right) \to \left(\widetilde{x}_0, 0\right), \ldots, T_{\vartheta_{N-1}} : \left(\widetilde{x}_0, x_{N-1}\right) \to \left(\widetilde{x}_0, 0\right) = (\pm\|\boldsymbol{x}\|, 0). \tag{15}$$

Here, at each step, the first value of the generator is updated as $\widetilde{x}_0$. The N-point DsiHT with this natural order of processing the components of the generators is called the weak DsiHT [26]. The transform of an input $\boldsymbol{z} = (z_0, z_1, \ldots, z_{N-1})\prime$ is processed with the same path,

$$T_{\vartheta_1} : (z_0, z_1) \to \left(\widetilde{z}_0, \widetilde{z}_1\right), T_{\vartheta_2} : \left(\widetilde{z}_0, z_2\right) \to \left(\widetilde{z}_0, \widetilde{z}_2\right), \ldots, T_{\vartheta_{N-1}} : \left(\widetilde{z}_0, z_{N-1}\right) \to \left(\widetilde{z}_0, \widetilde{z}_1\right). \tag{16}$$

Thus, the concept of the N-point DsiHT is based on creating a unitary transformation from a given signal or vector (or a multi-qubit superposition in quantum computation). The matrix of this transformation is composed of elementary rotations, and the rotation angles are determined by the generator and the path.

Example 1: Consider the generator $\boldsymbol{x} = (2, -1, 3, 4, 1, 2, 5, 1)/\sqrt{61}$. Seven rotations with angles $\{\vartheta_k,\ k = 1:7\} = \{26.56^\circ, -53.30^\circ, -46.91^\circ, -10.34^\circ, -19.75^\circ, -40.20^\circ, -7.35^\circ\}$ (in degrees) are used in the calculation of the 8-point weak DsiHT, which has the matrix

$$H_8 = \begin{bmatrix} 0.2561 & -0.1280 & 0.3841 & 0.5121 & 0.1280 & 0.2561 & 0.6402 & 0.1280 \\ 0.4472 & 0.8944 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.7171 & 0.3586 & 0.5976 & 0 & 0 & 0 & 0 & 0 \\ -0.3904 & 0.1952 & -0.5855 & 0.6831 & 0 & 0 & 0 & 0 \\ -0.0656 & 0.0328 & -0.0984 & -0.1312 & 0.9837 & 0 & 0 & 0 \\ -0.1214 & 0.0607 & -0.1822 & -0.2429 & -0.0607 & 0.9411 & 0 & 0 \\ -0.2182 & 0.1091 & -0.3273 & -0.4364 & -0.1091 & -0.2182 & 0.7638 & 0 \\ -0.0331 & 0.1265 & -0.0496 & -0.0661 & -0.0165 & -0.0331 & -0.0826 & 0.9918 \end{bmatrix}. \tag{17}$$

The determinant of this matrix is equal to 1, and the first row is the generator $\boldsymbol{x}$.

3. Decision Equations with Two Vector-Generators

In this section, we consider transformations which are defined by three decision equations with two generators. These are equations that allow us to calculate transformation parameters. Obviously, these equations must be solvable. The method of the composition of such transformations is based on the approach described in Section 2, for the DsiHT with one generator and two decision equations. Here, we consider the case when basic transformations $T_{\theta k}$ operate with only three components of an input vector $\boldsymbol{z} = (z_0, z_1, \ldots, z_{N-1})$. Namely, at each stage k of calculations, two components out of three from stage $(k-1)$ are updated, and the third one is chosen from the rest of $\boldsymbol{z}$. To avoid complex notations in equations, the same symbols z_k will be used for the original components of the input and for the updated ones. Thus, we consider an isomorphic 3D transformation $T_{\theta k}$ defined as follows:

$$\begin{gathered} T_{k_1,k_2,k_3}(\theta_k) : (z_{k_1}, z_{k_2}, z_{k_3}) \to (f_{k_1}(\boldsymbol{z}, \theta_k), f_{k_2}(\boldsymbol{z}, \theta_k), f_{k_3}(\boldsymbol{z}, \theta_k)) \\ = (f_{k_1}(z_{k_1}, z_{k_2}, z_{k_3}, \theta_k), f_{k_2}(z_{k_1}, z_{k_2}, z_{k_3}, \theta_k), f_{k_3}(z_{k_1}, z_{k_2}, z_{k_3}, \theta_k)) \\ k_1, k_2, k_3 \in \{0, 1, \ldots, N-1\}, \end{gathered} \tag{18}$$

where the triple of numbers (k_1, k_2, k_3) is uniquely defined by k. We also assume that the functions f_{k_1}, f_{k_2}, and f_{k_3} are equal to the same functions which we denote by f_1, f_2, and g, respectively. The problem of constructing an N-point transformation is reduced to defining 3-point basic transformations $T_{\theta_k} = T_{k_1,k_2,k_3}(\theta_k)$ as

$$T_{\theta_k} : (z_{k_1}, z_{k_2}, z_{k_3}) \to (f_1(z_{k_1}, z_{k_2}, z_{k_3}, \theta_k), f_2(z_{k_1}, z_{k_2}, z_{k_3}, \theta_k), g(z_{k_1}, z_{k_2}, z_{k_3}, \theta_k)). \tag{19}$$

Here, θ_k are vector parameters; for instance, $\theta_k = (\varphi_k, \psi_k)$, when $k = 1 : (N-2)$.

Rotations in 3D Space

To define a transformation generated by two vectors (generators), we consider two 2D rotations in the 3D space. Namely, we consider the performance of rotations around the x- and y-axes, and two given vector-generators will be rotated simultaneously to the plane x-y. In this case, the basic transformation $T_{\varphi,\psi}$ has the form

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \to \begin{bmatrix} f_1(x, y, z, \varphi, \psi) \\ f_2(x, y, z, \varphi, \psi) \\ g(x, y, z, \varphi, \psi) \end{bmatrix}, \tag{20}$$

where $\boldsymbol{v} = (x, y, z)$ is an input vector, and (φ, ψ) are angle parameters. These parameters are determined by two given vectors $\boldsymbol{v}_1 = (x_1, y_1, z_1)$ and $\boldsymbol{v}_2 = (x_2, y_2, z_2)$. Thus, the transformation is defined by three decision functions f_1, f_2, and g. We can also write $T_{\varphi,\psi} = T_{\boldsymbol{v}_1,\boldsymbol{v}_2}$.

It is assumed that parameters (φ, ψ) can be found from the angular equations

$$\begin{cases} g(x_1, y_1, z_1, \varphi, \psi) = a_1 \\ g(x_2, y_2, z_2, \varphi, \psi) = a_2 \end{cases} \tag{21}$$

for given parameters a_1 and a_2. The transformation $T_{\varphi,\psi}$ is defined by two rotations around the y- and x-axes,

$$T_{\varphi,\psi} = \begin{bmatrix} \cos\varphi & 0 & -\sin\varphi \\ 0 & 1 & 0 \\ \sin\varphi & 0 & \cos\varphi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{bmatrix} = \begin{bmatrix} \cos\varphi & -\sin\psi\sin\varphi & -\sin\varphi\cos\psi \\ 0 & \cos\psi & -\sin\psi \\ \sin\varphi & \sin\psi\cos\varphi & \cos\psi\cos\varphi \end{bmatrix}. \tag{22}$$

The angular equations for the vector-generators $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ can be written as

$$\begin{cases} x_1\sin\varphi + y_1\sin\psi\cos\varphi + z_1\cos\psi\cos\varphi = a_1 \\ x_2\sin\varphi + y_2\sin\psi\cos\varphi + z_2\cos\psi\cos\varphi = a_2 \end{cases}. \tag{23}$$

We consider the particular case when $a_1 = a_2 = 0$. It is not difficult to see that angles φ and ψ, being vector functions of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, are determined from the above angular equations as follows:

$$\begin{array}{c} \tan\psi = \frac{x_2z_1 - z_2x_1}{y_2x_1 - x_2y_1}, \\ \tan\varphi = -\frac{1}{x_1} \cdot \frac{y_1\tan\psi + z_1}{\sqrt{1+\tan^2\psi}}, \end{array} \tag{24}$$

and $\varphi = \pi/2$ if $x_1 = 0$.

To compose the transform of the N-dimension vector $\boldsymbol{z} = (z_0, z_1, \ldots, z_{N-1})$, we consider two vector-generators $\boldsymbol{x} = (x_0, x_1, \ldots, x_{N-1})$ and $\boldsymbol{y} = (y_0, y_1, \ldots, y_{N-1})$. The order of choosing all triplets (k_1, k_2, k_3) in Equation (19) is called the path of the D2siHT. We consider the case when the path of the transformation H is defined in the natural order, from left to right, as $0, 1, 2, \ldots, (N-1)$. In other words, let us define the following triplets:

$$\begin{array}{ccc} \boldsymbol{x}_1 = (x_0, x_1, x_2), & \boldsymbol{y}_1 = (y_0, y_1, y_2), & \boldsymbol{z}_1 = (z_0, z_1, z_2), \\ \boldsymbol{x}_2 = \left(\widetilde{x}_0, \widetilde{x}_1, x_3\right), & \boldsymbol{y}_2 = \left(\widetilde{y}_0, \widetilde{y}_1, y_3\right), & \boldsymbol{z}_2 = \left(\widetilde{z}_0, \widetilde{z}_1, z_3\right), \\ \boldsymbol{x}_3 = \left(\widetilde{x}_0, \widetilde{x}_1, x_4\right), & \boldsymbol{y}_3 = \left(\widetilde{y}_0, \widetilde{y}_1, y_4\right), & \boldsymbol{z}_3 = \left(\widetilde{z}_0, \widetilde{z}_1, z_4\right), \\ \ldots & \ldots & \ldots \\ \boldsymbol{x}_k = \left(\widetilde{x}_0, \widetilde{x}_1, x_{k+1}\right), & \boldsymbol{y}_k = \left(\widetilde{y}_0, \widetilde{y}_1, y_{k+1}\right), & \boldsymbol{z}_k = \left(\widetilde{z}_0, \widetilde{z}_1, z_{k+1}\right), \end{array} \tag{25}$$

when $k = 2 : (N-2)$. The first two components of each triplet are updated from the $(k-1)$-th stage as

$$\begin{array}{c} \left(\widetilde{x}_0, \widetilde{x}_1, 0\right)' = T_{\varphi_k,\psi_k}\left(\widetilde{x}_0, \widetilde{x}_1, x_{k+1}\right)' \\ \left(\widetilde{y}_0, \widetilde{y}_1, 0\right)' = T_{\varphi_k,\psi_k}\left(\widetilde{y}_0, \widetilde{y}_1, y_{k+1}\right)' \\ \left(\widetilde{z}_0, \widetilde{z}_1, z_k\right)' = T_{\varphi_k,\psi_k}\left(\widetilde{z}_0, \widetilde{z}_1, z_{k+1}\right)'. \end{array} \tag{26}$$

Before the last stage of the transform calculation, that is, when $k = (N-2)$, we obtain the following transforms of the generators:

$$\begin{array}{c} \boldsymbol{x} = (x_0, x_1, \ldots, x_{N-1}) \to \hat{\boldsymbol{x}} = \left(\widetilde{x}_0, \widetilde{x}_1, 0, \ldots, 0\right), \\ \boldsymbol{y} = (y_0, y_1, \ldots, y_{N-1}) \to \hat{\boldsymbol{y}} = \left(\widetilde{y}_0, \widetilde{y}_1, 0, \ldots, 0\right), \end{array} \tag{27}$$

and for the input, $\boldsymbol{z} = (z_0, z_1, \ldots, z_{N-1}) \to \left(\widetilde{z}_0, \widetilde{z}_1, \widetilde{z}_2, \ldots, \widetilde{z}_{N-1}\right)$. Our goal is to transfer these two vectors into the first and second Cartesian unit vectors,

$$\hat{\boldsymbol{x}} \to (1, 0, 0, \ldots, 0) \text{ and } \hat{\boldsymbol{y}} \to (0, 1, 0, \ldots, 0). \tag{28}$$

Therefore, the transformation D which is applied to the two-dimension vector $\left(\widetilde{z}_0, \widetilde{z}_1\right)'$ has the following matrix:

$$D = \frac{1}{\widetilde{x}_0\widetilde{y}_1 - \widetilde{y}_0\widetilde{x}_1}\begin{bmatrix} \widetilde{y}_1 & -\widetilde{y}_0 \\ -\widetilde{x}_1 & \widetilde{x}_0 \end{bmatrix}, \det D = \frac{1}{\widetilde{x}_0\widetilde{y}_1 - \widetilde{y}_0\widetilde{x}_1} \tag{29}$$

Indeed, it is not difficult to see that

$$D\begin{bmatrix} \widetilde{x}_0 \\ \widetilde{x}_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } D\begin{bmatrix} \widetilde{y}_0 \\ \widetilde{y}_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{30}$$

It is assumed that the vectors $\left(\widetilde{x}_0, \widetilde{x}_1\right)$ and $\left(\widetilde{y}_0, \widetilde{y}_1\right)$ are independent. At this stage, the first two components of the transform $\left(\widetilde{z}_0, \widetilde{z}_1, \widetilde{z}_2, \ldots, \widetilde{z}_{N-1}\right)$ are processed by this matrix, $\left(\widetilde{z}_0, \widetilde{z}_1\right)' \to D\left(\widetilde{z}_0, \widetilde{z}_1\right)'$. The N-point D2siHT of the input signal is

$$H\boldsymbol{z} = \left(\left(\widetilde{z}_0, \widetilde{z}_1\right)\prime D\prime, \widetilde{z}_2, \ldots, \widetilde{z}_{N-1}\right). \tag{31}$$

Thus, the discrete two signal-induced heap transformation (D2siHT) composed of the basic transformations (rotations) $T_{\varphi,\psi}$ is defined as

$$H\boldsymbol{z} = (D \circ T)z = D \circ \underbrace{T_{\varphi_{N-2},\psi_{N-2}}(\boldsymbol{z}_{N-2}) \circ \ldots \circ T_{\varphi_2,\psi_2}(\boldsymbol{z}_2) \circ T_{\varphi_1,\psi_1}(\boldsymbol{z}_1)}, \tag{32}$$

where the vector triples $\boldsymbol{z}_k$, $k = 1 : (N-2)$, and angles φ_k and ψ_k are determined based on the processing of $\boldsymbol{z}$ data and the generators $\boldsymbol{x}_k$ and $\boldsymbol{y}_k$.

Comment 1: If the vectors $\left(\widetilde{x}_0, \widetilde{x}_1\right)$ and $\left(\widetilde{y}_0, \widetilde{y}_1\right)$ at the last stage of calculation in Equation (27) are dependent, then $\left(\widetilde{y}_0, \widetilde{y}_1\right) = \pm\left(\widetilde{x}_0, \widetilde{x}_1\right)$ when $\|\boldsymbol{x}\| = \|\boldsymbol{y}\| = 1$. In this case, the matrix D is considered to be the elementary rotation, and

$$D\begin{bmatrix} \widetilde{x}_0 \\ \widetilde{x}_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } D\begin{bmatrix} \widetilde{y}_0 \\ \widetilde{y}_1 \end{bmatrix} = \pm\begin{bmatrix} 1 \\ 0 \end{bmatrix}. \tag{33}$$

Two generators $\boldsymbol{x}$ and $\boldsymbol{y}$ are transformed to the vectors $\pm(1, 0, 0, \ldots, 0)$.

Comment 2: We may assume that the vector-generators are unit vectors; that is, $\|\boldsymbol{x}\| = 1$, and $\|\boldsymbol{y}\| = 1$. Then, the transforms in Equation (28) can be written as $\hat{\boldsymbol{x}} \to (\|\boldsymbol{x}\|, 0, 0, \ldots, 0)$ and $\hat{\boldsymbol{y}} \to (0, \|\boldsymbol{y}\|, 0, \ldots, 0)$. In the case, when $\|\boldsymbol{x}\| \neq 1$ and $\|\boldsymbol{y}\| \neq 1$, such a matrix D does not exist. In other words, at the last stage of the DsiHT, it is not possible to find a matrix D such that

$$D\begin{bmatrix} \widetilde{x}_0 \\ \widetilde{x}_1 \end{bmatrix} = \begin{bmatrix} \|\boldsymbol{x}\| \\ 0 \end{bmatrix} \text{ and } D\begin{bmatrix} \widetilde{y}_0 \\ \widetilde{y}_1 \end{bmatrix} = \begin{bmatrix} 0 \\ \|\boldsymbol{y}\| \end{bmatrix}. \tag{34}$$

Comment 3: All matrices of the transformations in Equation (32), except the matrix D, are rotation matrices. Therefore, the D2siHT is not unitary, and its matrix, H, has determinant $\det H = \det D$. To apply the D2siHT in quantum computation, for the case

when N is a power of 2, we will replace the matrix D with a rotation around the z-axis. One of the following transforms is considered:

$$\hat{\boldsymbol{x}} \to (\|\boldsymbol{x}\|, 0, 0, \ldots, 0) \text{ and } \hat{\boldsymbol{y}} \to \left(\left(\widetilde{y}_0, \widetilde{y}_1 \right)' R_{\vartheta_1}, 0, \ldots, 0 \right) \tag{35}$$

and

$$\hat{\boldsymbol{y}} \to (0, \|\boldsymbol{y}\|, \ldots, 0) \text{ and } \hat{\boldsymbol{x}} \to \left(\left(\widetilde{x}_0, \widetilde{x}_1 \right)' R_{\vartheta_2}, 0, \ldots, 0 \right). \tag{36}$$

Here, R_{ϑ_1} and R_{ϑ_2} are the matrices of elementary rotations (the Givens rotations) defined as

$$R_{\vartheta_1} \begin{bmatrix} \widetilde{x}_0 \\ \widetilde{x}_1 \end{bmatrix} = \begin{bmatrix} \cos\vartheta_1 & -\sin\vartheta_1 \\ \sin\vartheta_1 & \cos\vartheta_1 \end{bmatrix} \begin{bmatrix} \widetilde{x}_0 \\ \widetilde{x}_1 \end{bmatrix} = \begin{bmatrix} \|\boldsymbol{x}\| \\ 0 \end{bmatrix} \tag{37}$$

and

$$R_{\vartheta_2} \begin{bmatrix} \widetilde{y}_0 \\ \widetilde{y}_1 \end{bmatrix} = \begin{bmatrix} \cos\vartheta_2 & -\sin\vartheta_2 \\ \sin\vartheta_2 & \cos\vartheta_2 \end{bmatrix} \begin{bmatrix} \widetilde{y}_0 \\ \widetilde{y}_1 \end{bmatrix} = \begin{bmatrix} 0 \\ \|\boldsymbol{y}\| \end{bmatrix}. \tag{38}$$

The angles are calculated by

$$\vartheta_1 = -\arctan\left(\widetilde{x}_1 / \widetilde{x}_0 \right), \text{and } \vartheta_1 = \pi/2 \text{ if } \widetilde{x}_0 = 0, \tag{39}$$

$$\vartheta_2 = \arctan\left(\widetilde{y}_0 / \widetilde{y}_1 \right), \text{and } \vartheta_2 = \pi/2 \text{ if } \widetilde{y}_1 = 0. \tag{40}$$

The normalization of the generators, $\boldsymbol{x} \to \boldsymbol{x}/\|\boldsymbol{x}\|$ and $\boldsymbol{y} \to \boldsymbol{y}/\|\boldsymbol{y}\|$, is not required. Thus, the modified N-point D2siHT is considered as the transform

$$H\boldsymbol{z} = (R_\vartheta \circ T)\boldsymbol{z} = R_\vartheta \circ \underbrace{T_{\varphi_{N-2},\psi_{N-2}}(\boldsymbol{z}_{N-2}) \circ \ldots \circ T_{\varphi_2,\psi_2}(\boldsymbol{z}_2) \circ T_{\varphi_1,\psi_1}(\boldsymbol{z}_1)}, \tag{41}$$

where $\vartheta = \vartheta_1$ or ϑ_2. This can also be written as $H\boldsymbol{z} = \left(\left(\widetilde{z}_0, \widetilde{z}_1 \right)' R_{-\vartheta}, \widetilde{z}_2, \ldots, \widetilde{z}_{N-1} \right)$. The transformation is unitary and uses $(N-2)$ 3D rotations and one 2D rotation. All angles of this transformation can be written as shown in Table 1. Only one of the angles ϑ_1 and ϑ_2 is used. The total number of angles in the D2siHT is equal to $2(N-2)+1 = 2N-3$.

Table 1. Encoding data of angles for D2siHT.

R_y	φ_1	φ_2	$\cdots$	φ_{N-3}	φ_{N-2}
R_x	ψ_1	ψ_2	$\cdots$	ψ_{N-3}	ψ_{N-2}
R_z	ϑ_1	ϑ_2			

Example 2: ($N = 8$) Consider two generators $\boldsymbol{x} = (1, -2, 4, 5, -2, 5, 1, 3)'$ and $\boldsymbol{y} = (2, 7, -6, 4, 1, -2, 5, 2)'$. The D2siHT has a matrix calculated by

$$H_8 = R_{z;\vartheta_1}(0,1)T = R_{z;\vartheta_1}(0,1)\prod_{k=0}^{5}\left(R_{y;\varphi_{7-k}}(0, 7-k) R_{x;\psi_{7-k}}(1, 7-k) \right), \tag{42}$$

and it is equal to

$$H_8 = \begin{bmatrix} 0.1085 & -0.2169 & 0.4339 & 0.5423 & -0.2169 & 0.5423 & 0.1085 & 0.3254 \\ 0.1889 & 0.5668 & -0.4466 & 0.4294 & 0.0515 & -0.0859 & 0.4466 & 0.2233 \\ -0.6684 & 0.5849 & 0.4595 & 0 & 0 & 0 & 0 & 0 \\ -0.6900 & -0.4139 & -0.4769 & 0.3539 & 0 & 0 & 0 & 0 \\ 0.0348 & -0.1019 & 0.1803 & 0.1916 & 0.9588 & 0 & 0 & 0 \\ -0.0778 & 0.1736 & -0.3340 & -0.3987 & 0.1637 & 0.8164 & 0 & 0 \\ -0.2139 & -0.2723 & 0.1664 & -0.3358 & 0.0114 & -0.0521 & 0.8759 & 0 \\ -0.0843 & -0.0609 & -0.0451 & -0.2964 & 0.0643 & -0.1712 & -0.1469 & 0.9188 \end{bmatrix}. \quad (43)$$

Here, for axes $k = x, y$, and z, the notation $R_{k,\vartheta}(i,j)$, is used for the rotation gate $R_{k,\vartheta}$ that operates on the biplanes i and j, $i \neq j$. All angles for this 8-point transformation are given in Table 2.

Table 2. Encoding data of angles for 8-point D2siHT.

R_y :	$-41.94°$	$-68.07°$	$7.19°$	$-16.39°$	$-27.94°$	$-21.16°$
R_x :	$51.28°$	$18.62°$	$-14.90°$	$31.68°$	$-7.48°$	$9.85°$
R_z :	$60.14°$	$-38.86°$				

The transforms of the generators are

$$\begin{gathered} H_8\boldsymbol{x} = (\|\boldsymbol{x}\|, 0, 0, \ldots, 0)' = (9.2195, 0, 0, \ldots, 0)', \\ H_8\boldsymbol{y} = (-1.8439, 11.6447, 0, \ldots, 0)'. \end{gathered} \quad (44)$$

The inverse transformation is defined by the matrix

$$H_8' = T' R_{z;-\vartheta_1}(0,1) = \left[\prod_{k=2}^{7} \left(R_{x;-\psi_k}(1,k) R_{y;-\varphi_k}(0,k)\right)\right] R_{z;-\vartheta_1}(0,1). \quad (45)$$

If in the last rotation around the z-axis, we use the angle $\vartheta_2 = -38.86°$ instead of $\vartheta_1 = 60.14°$, we obtain a D2siHT with the matrix $H_{8;2}$, in which only the first two rows differ from the matrix in Equation (43),

$$H_{8;2} = \begin{bmatrix} 0.1696 & 0.5937 & -0.5089 & 0.3393 & 0.0848 & -0.1696 & 0.4221 & 0.1696 \\ -0.1367 & 0.1256 & -0.3587 & -0.6028 & 0.2062 & -0.5222 & -0.1770 & -0.3563 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}. \quad (46)$$

Then, the transforms of the generators are

$$\begin{gathered} H_{8;2}\boldsymbol{y} = (\|\boldsymbol{y}\|, 0, 0, \ldots, 0)' = (11.7898, 0, 0, \ldots, 0)', \\ H_{8;2}\boldsymbol{x} = (-1.4419, -9.1061, 0, \ldots, 0)'. \end{gathered} \quad (47)$$

If we use the D2siHT in the original form, that is, with the matrix D,

$$D = \begin{bmatrix} 0.0689 & 0.0745 \\ -0.0855 & 0.0428 \end{bmatrix}, \ \det D = 0.0093, \quad (48)$$

then the matrix of the transform calculated by

$$H_{8;3} = D(0,1) \prod_{k=0}^{5} \left(R_{y;\varphi_{7-k}}(0, 7-k) R_{x;\psi_{7-k}}(1, 7-k)\right) \quad (49)$$

changes only in the first two rows in the matrix H_8 in Equation (43), as shown below

$$H_{8;3} = \begin{bmatrix} 0.0150 & -0.0138 & 0.0394 & 0.0662 & -0.0226 & 0.0573 & 0.0194 & 0.0391 \\ 0.0162 & 0.0487 & -0.0383 & 0.0369 & 0.0044 & -0.0074 & 0.0383 & 0.0192 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}. \quad (50)$$

This transformation is not unitary and over the generators results in the unit vectors $H_{8;3}\boldsymbol{x} = (1,0,0,\ldots,0)'$ and $H_{8;3}\boldsymbol{y} = (0,1,0,\ldots,0)'$.

Example 3: For the $N = 16$ case, we can represent two generators $\boldsymbol{x}$ and $\boldsymbol{y}$ as 4-qubit superpositions

$$|\boldsymbol{x}\rangle = x_0|0\rangle + x_1|1\rangle + \cdots + x_{15}|15\rangle,\ |\boldsymbol{y}\rangle = y_0|0\rangle + y_1|1\rangle + \cdots + y_{15}|15\rangle, \quad (51)$$

and the input $\boldsymbol{z}$ as $|\boldsymbol{z}\rangle = z_0|0\rangle + z_1|1\rangle + \cdots + z_{15}|15\rangle$. Here, $|k\rangle$, $k = 0:15$, denote the computational basis states [27]. The 4-qubit Q2siHT is described by the matrix of the 16-point D2siHT, H_{16}, generated by $\boldsymbol{x}$ and $\boldsymbol{y}$. The circuit for calculating this transform is given in Figure 2, and the circuit for calculating the inverse 4-qubit Q2siHT is given in Figure 3. The 2D rotation R_ϑ operates on bit planes 0 and 1. Therefore, this rotation is shown in the diagram as three zero-controlled rotation gate.

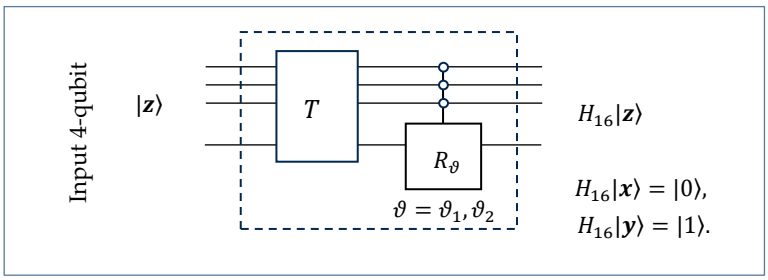


Figure 2. A diagram of the 4-qubit Q2siHT.

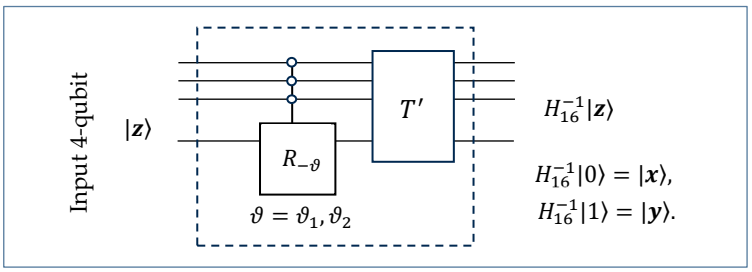


Figure 3. A diagram of the inverse 4-qubit Q2siHT.

Example 4: The 4-point D2siHT is described by the following matrix:

$$H_4 = (R_\vartheta \oplus I_2)T_{\varphi_2,\psi_2}T_{\varphi_1,\psi_1} = (R_\vartheta \oplus I_2)[R_3(\varphi_2)R_4(\psi_2)][R_1(\varphi_1)R_2(\psi_1)], \quad (52)$$

where

$$T_{\varphi_1,\psi_1} = R_1(\varphi_1)R_2(\psi_1) = \begin{bmatrix} \cos\varphi_1 & 0 & -\sin\varphi_1 & 0 \\ 0 & 1 & 0 & 0 \\ \sin\varphi_1 & 0 & \cos\varphi_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\psi_1 & -\sin\psi_1 & 0 \\ 0 & \sin\psi_1 & \cos\psi_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (53)$$

$$T_{\varphi_2,\psi_2} = R_3(\varphi_2)R_4(\psi_2) = \begin{bmatrix} \cos\varphi_2 & 0 & 0 & -\sin\varphi_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \sin\varphi_2 & 0 & 0 & \cos\varphi_2 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\psi_2 & 0 & -\sin\psi_2 \\ 0 & 0 & 1 & 0 \\ 0 & \sin\psi_2 & 0 & \cos\psi_2 \end{bmatrix}, \quad (54)$$

$$R_\vartheta \oplus I_2 = \begin{bmatrix} \cos\vartheta & -\sin\vartheta & 0 & 0 \\ \sin\vartheta & \cos\vartheta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (55)$$

The rotations $R_2(\psi_1)$ and $R_3(\varphi_2)$ operate on non-adjacent bit planes. We need to move these rotations on adjacent bit planes. For this, we consider the matrix of permutation $P_{0,1} = (0,1)$ and write the matrices of these rotations as

$$R_2(\psi_1) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\psi_1 & 0 & -\sin\psi_1 & 0 \\ 0 & 1 & 0 & 0 \\ \sin\psi_1 & 0 & \cos\psi_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (56)$$

$$R_3(\varphi_2) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\varphi_2 & 0 & -\sin\varphi_2 \\ 0 & 0 & 1 & 0 \\ 0 & \sin\varphi_2 & 0 & \cos\varphi_2 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (57)$$

Thus,

$$H_4 = (R_\vartheta \oplus I_2)[\underbrace{P_{0,1}R_4(\varphi_2)P_{0,1}}\, R_4(\psi_2)][R_1(\varphi_1)\, \underbrace{P_{0,1}R_1(\psi_1)P_{0,1}}]. \quad (58)$$

The circuit of this transformation is given in Figure 4. The permutation $(0,1)$ represents the 0-controlled X gate or NOT gate; that is, $P_{0,1} = X \oplus I_2$ [28].

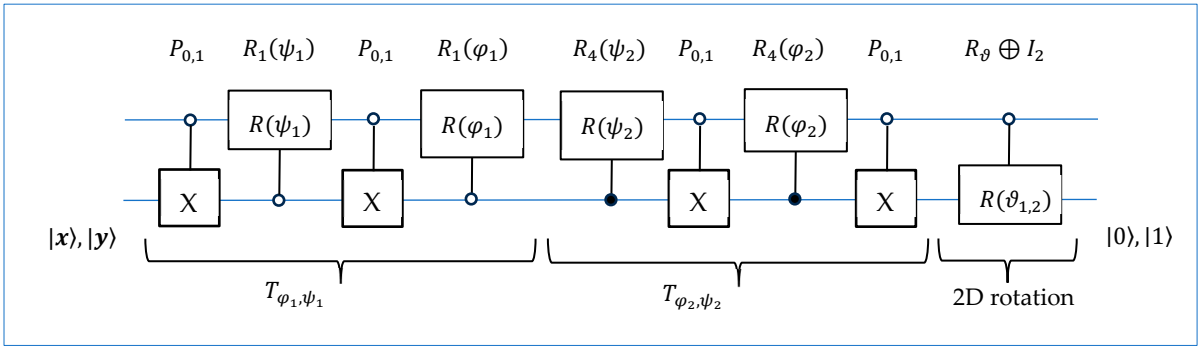


Figure 4. The circuit for the 2-qubit D2siHT.

The quantum circuit for the inverse transformation is given in Figure 5. This circuit can be used to initiate the 2-qubit superpositions $|x\rangle$ and $|y\rangle$ from the states $|0\rangle$ and $|1\rangle$, respectively. Namely, the state $|x\rangle = H_4|0\rangle$ when ϑ=ϑ_1, and the state $|y\rangle = H_4|1\rangle$, when ϑ=ϑ_2.

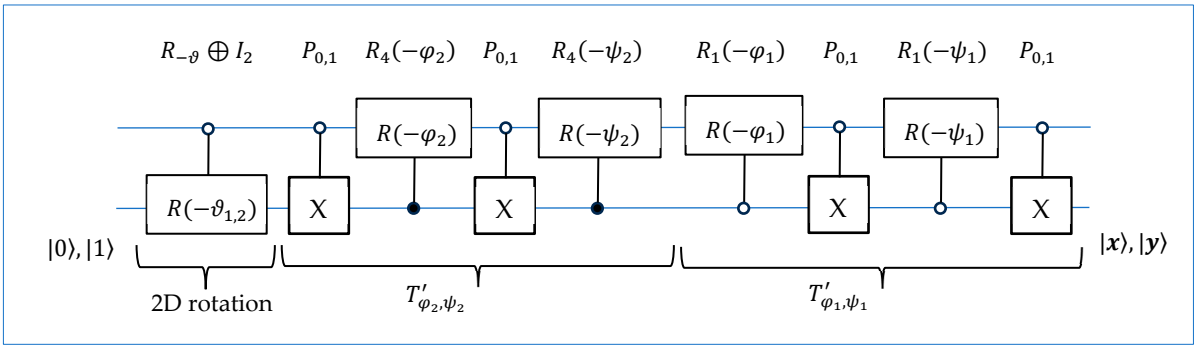


Figure 5. The circuit for the preparation of two superpositions $|x\rangle$ and $|y\rangle$.

Let us consider, for example, the following 2-qubit superpositions:

$$|\boldsymbol{x}\rangle = \frac{-2|0\rangle + 3|1\rangle + |2\rangle + 4|3\rangle}{\sqrt{30}} \text{ and } |\boldsymbol{y}\rangle = \frac{|0\rangle + 4|1\rangle - 5|2\rangle + 2|3\rangle}{\sqrt{46}}. \tag{59}$$

The angles of the transformation H_4 are given in Table 3.

Table 3. Encoding data of angles for 2-qubit Q2siHT.

R_y :	$\varphi_1 = 53.20^\circ$	$\varphi_2 = 38.52^\circ$
R_x :	$\psi_1 = 39.29^\circ$	$\psi_2 = -29.37^\circ$
R_z :	$\vartheta_1 = 218.82^\circ$	$\vartheta_2 = -71.66^\circ$

The matrices of the D2siHTs are

$$H_{4;\vartheta_1} = \begin{bmatrix} -0.3651 & 0.5477 & 0.1826 & 0.7303 \\ -0.2938 & -0.4250 & 0.8552 & -0.0420 \\ 0.8008 & 0.3793 & 0.4636 & 0 \\ 0.3730 & -0.6128 & -0.1429 & 0.6818 \end{bmatrix}, \tag{60}$$

$$H_{4;\vartheta_2} = \begin{bmatrix} 0.1474 & 0.5898 & -0.7372 & 0.2949 \\ -0.4449 & 0.3644 & 0.4703 & 0.6694 \\ 0.8008 & 0.3793 & 0.4636 & 0 \\ 0.3730 & -0.6128 & -0.1429 & 0.6818 \end{bmatrix}. \tag{61}$$

The transforms of the generators are

$$\begin{aligned} &H_{4;\vartheta_1}\boldsymbol{x} = (1,0,0,0)', H_{4;\vartheta_1}\boldsymbol{y} = (0.3499, -0.9368, 0, 0)', \\ &H_{4;\vartheta_2}\boldsymbol{y} = (0,1,0,0)', H_{4;\vartheta_2}\boldsymbol{x} = (0.3499, 0.9368, 0, 0)'. \end{aligned} \tag{62}$$

Figure 6 shows the roadmap of this transformation. Five rotations in this transformation are marked with crosses, which we call butterflies. Each 3D rotation is shown as a pair of butterflies. The zero outputs of these butterflies are marked with open circles. The red circles represent the points at which the signal energy is transformed, or the last non-zero outputs of the generator transformations. The butterflies marked with a red dashed circles operate on non-adjacent bit planes. Finding a path with three adjacent bit planes for all 3D rotations T_{φ_k,ψ_k} is a difficult task and probably impossible.

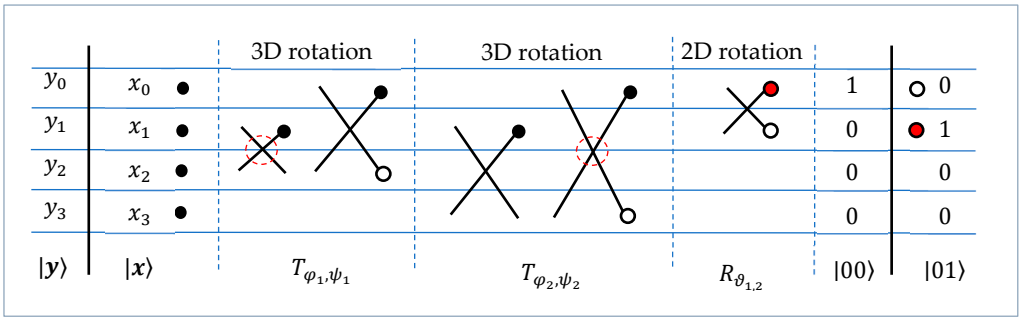


Figure 6. The roadmap of the 4-point D2siHT.

Now, we consider the roadmap that is similar to the concept of the strong DsiHT with one generator [25,26]. As shown in Figure 7, the data are processed starting from the last three components of two generators $\boldsymbol{x}$ and $\boldsymbol{y}$.

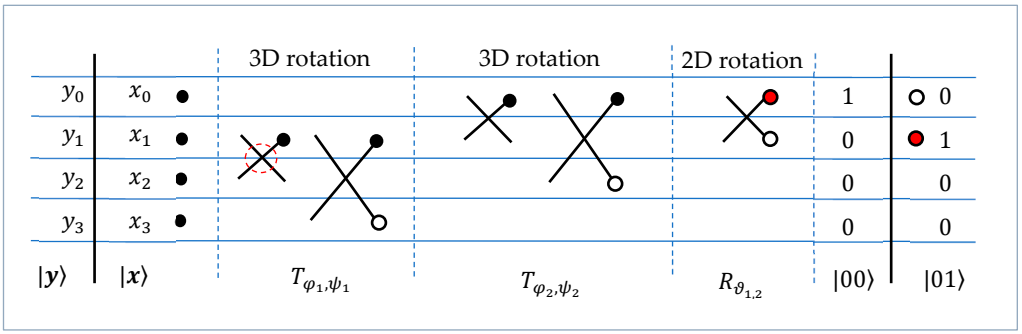


Figure 7. The roadmap of the 4-point strong D2siHT.

For the above generators, the angles of this transformation are given in Table 4.

Table 4. Encoding data of angles for 2-qubit strong Q2siHT.

R_y :	$\varphi_1 = -45.69^\circ$	$\varphi_2 = 62.05^\circ$
R_x :	$\psi_1 = -27.76^\circ$	$\psi_2 = 15.06^\circ$
R_z :	$\vartheta_1 = 218.82^\circ$	$\vartheta_2 = 18.34^\circ$

The matrix of the transformation is equal to

$$H_{4;\vartheta_1} = \begin{bmatrix} -0.3651 & 0.5477 & 0.1826 & 0.7303 \\ -0.2938 & -0.4250 & 0.8552 & -0.0420 \\ 0.8834 & 0.0851 & 0.3599 & 0.2879 \\ 0 & -0.7157 & -0.3253 & 0.6181 \end{bmatrix}. \quad (63)$$

4. The 3-Qubit Q2siHT

For large values of N, many different roadmaps can be used for the N-point D2siHT. We describe such roadmaps for the $N = 8$ case. First, we consider the roadmap for the 8-point D2siHT (or the 3-qubit Q2siHT) that is used in Example 2. It is shown in Figure 8. One can note that with each step of calculation, control (that is, the butterfly operation) is carried out with increasingly distant signal components, which we attribute to "the weakness" of such a path in the roadmap. This is why we call the transformation with such a path, or path #1, the weak D2siHT.

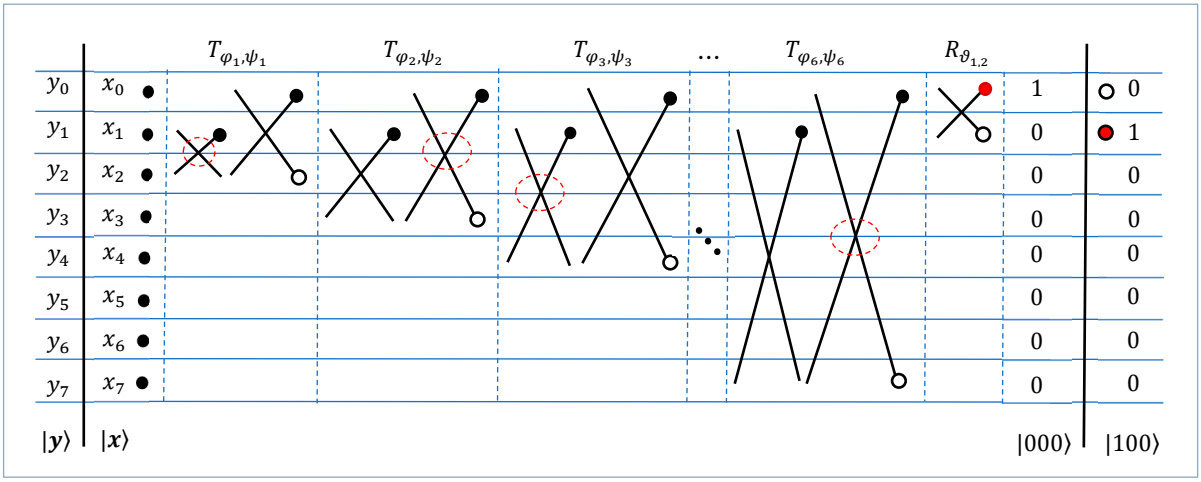


Figure 8. The roadmap of the 8-point D2siHT with the natural path or path #1.

Figure 9 shows the roadmap for the 8-point D2siHT, which we call a strong D2siHT. The path of processing data is different. With each step of calculation, the butterflies operate on the nearest component signals, which we consider to be a strong characteristic of such a path. This path will be called path #2.

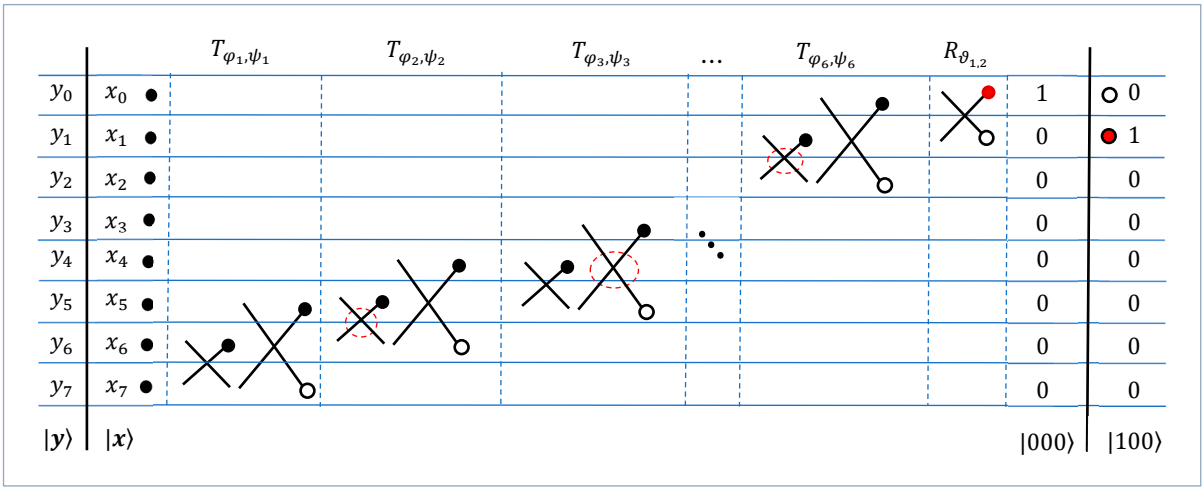


Figure 9. The roadmap of the 8-point D2siHT with path #2.

It should be noted that when using path #1 and path #2, there are 3D butterflies whose two 2D butterflies operate on non-disjoint BPs. For instance, when using path #1, the first butterfly in T_{φ_5,ψ_5} operates on BPs 001 and 110, and the second one operates on BPs 000 and 110. When using path #2, such a 3D butterfly is T_{φ_4,ψ_4} which operates on BPs 011 and 100 and then on 010 and 100.

Example 5: Let vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ be two generators considered in Example 2. Then, the 8-point strong D2siHT is described by rotations with the angles given in Table 5.

Table 5. Encoding data of angles for 8-point strong D2siHT.

R_y :	$-22.50°$	$-68.94°$	$-43.09°$	$40.05°$	$-52.20°$	$-77.41°$
R_x :	$-30.65°$	$-85.25°$	$-21.56°$	$6.28°$	$-71.00°$	$39.87°$
R_z :	$60.14°$	$51.14°$				

The matrix of this transformation is equal to

$$H_{8;2} = \begin{bmatrix} 0.1085 & -0.2169 & 0.4339 & 0.5423 & -0.2169 & 0.5423 & 0.1085 & 0.3254 \\ 0.1889 & 0.5668 & -0.4466 & 0.4294 & 0.0515 & -0.0859 & 0.4466 & 0.2233 \\ -0.9760 & 0.0856 & -0.0382 & 0.1434 & -0.0141 & 0.0436 & 0.0985 & 0.0794 \\ 0 & -0.7902 & -0.4436 & 0.1746 & 0.0950 & -0.2058 & 0.3012 & 0.0794 \\ 0 & 0 & 0.6435 & 0.0612 & 0.2466 & -0.5644 & 0.4502 & -0.0050 \\ 0 & 0 & 0 & -0.6832 & -0.0964 & 0.2825 & 0.5048 & 0.4352 \\ 0 & 0 & 0 & 0 & -0.9332 & -0.3308 & 0.0954 & -0.1027 \\ 0 & 0 & 0 & 0 & 0 & -0.3827 & -0.4710 & 0.7948 \end{bmatrix}. \quad (64)$$

As an example, Figure 10 shows another roadmap of the 8-point transformation.

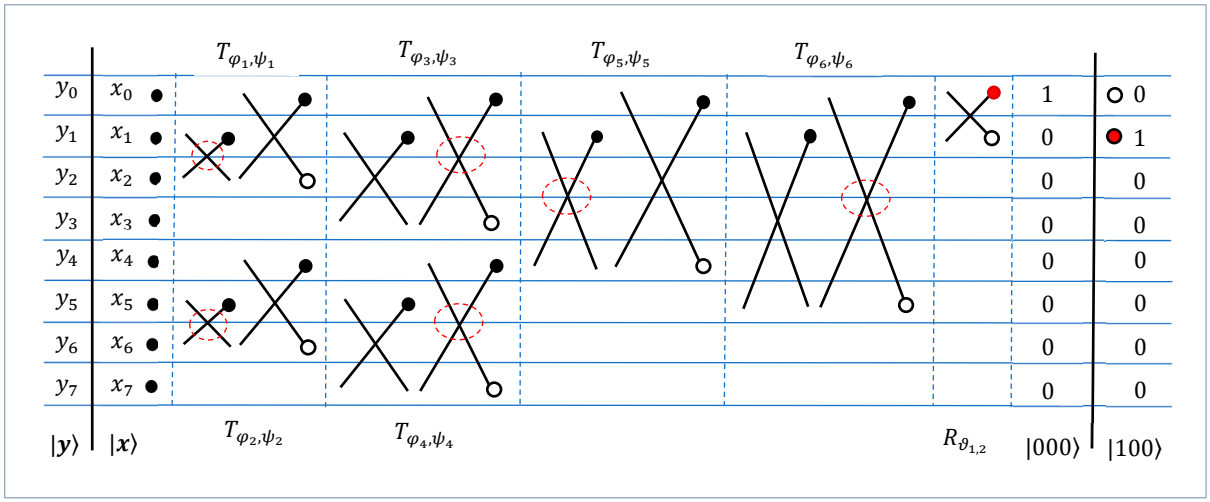


Figure 10. The 3rd roadmap of the 8-point D2siHT.

The path with partitioning, or path #3, in this map with five steps of calculations can be considered more effective than path #1, which is used in Example 2. It is not difficult to see that in this roadmap, in all 3D butterflies, only one of two 2D butterflies operates on non-disjoint BPs. This is also true in the general case of $N = 2^r$, $r > 2$. The number of stages or the depth of the circuit corresponding to this roadmap is equal to $2(r-1)+1$. For the roadmaps with paths #1 and #2, the number of stages in the calculations is equal to $(N-2)+1$.

We consider the same two generators $\boldsymbol{x}$ and $\boldsymbol{y}$ that are used in Example 2. The matrix of the 8-point D2siHT with path #3 is equal to

$$H_{8;3} = \begin{bmatrix} 0.1085 & -0.2169 & 0.4339 & 0.5423 & -0.2169 & 0.5423 & 0.1085 & 0.3254 \\ 0.1889 & 0.5668 & -0.4466 & 0.4294 & 0.0515 & -0.0859 & 0.4466 & 0.2233 \\ -0.6684 & 0.5849 & 0.4595 & 0 & 0 & 0 & 0 & 0 \\ -0.6900 & -0.4139 & -0.4769 & 0.3539 & 0 & 0 & 0 & 0 \\ 0.0779 & -0.2281 & 0.4037 & 0.4290 & 0.2771 & -0.6627 & 0.1912 & -0.2058 \\ -0.1538 & -0.2575 & 0.1041 & -0.4606 & 0.0507 & -0.0588 & 0.7220 & 0.3996 \\ 0 & 0 & 0 & 0 & -0.9255 & -0.3771 & 0.0343 & 0 \\ 0 & 0 & 0 & 0 & 0.1196 & -0.3371 & -0.4793 & 0.8014 \end{bmatrix}. \quad (65)$$

The matrix changed only in the second part, when compared with the matrix H_8 in Equation (43). The number of zero coefficients is 18, compared to 15 in H_8 and $H_{8;2}$. This is also true in the general case of generators. Each path defines the structure of the transformation matrix. All angles for this 8-point transformation are given in Table 6.

Table 6. Encoding data of angles for 8-point D2siHT with path #3.

R_y :	$-41.94°$	$-67.75°$	$-68.07°$	$18.41°$	$16.29°$	$-35.22°$
R_x :	$51.28°$	$-84.81°$	$18.62°$	$-32.36°$	$-36.53°$	$1.31°$
R_z :	$60.14°$	$-51.14°$				

The quantum circuit for the transformation is given in Figure 11. This circuit is composed of 13 controlled rotation gates and 12 controlled NOT gates. All these gates are controlled by two qubits. In the general case, the number of rotation gates to initiate r-qubit superposition is equal to $n(R) = 2(2^r - 1)$ plus $n(X) = 2(2^r - 2)$ controlled X gates.

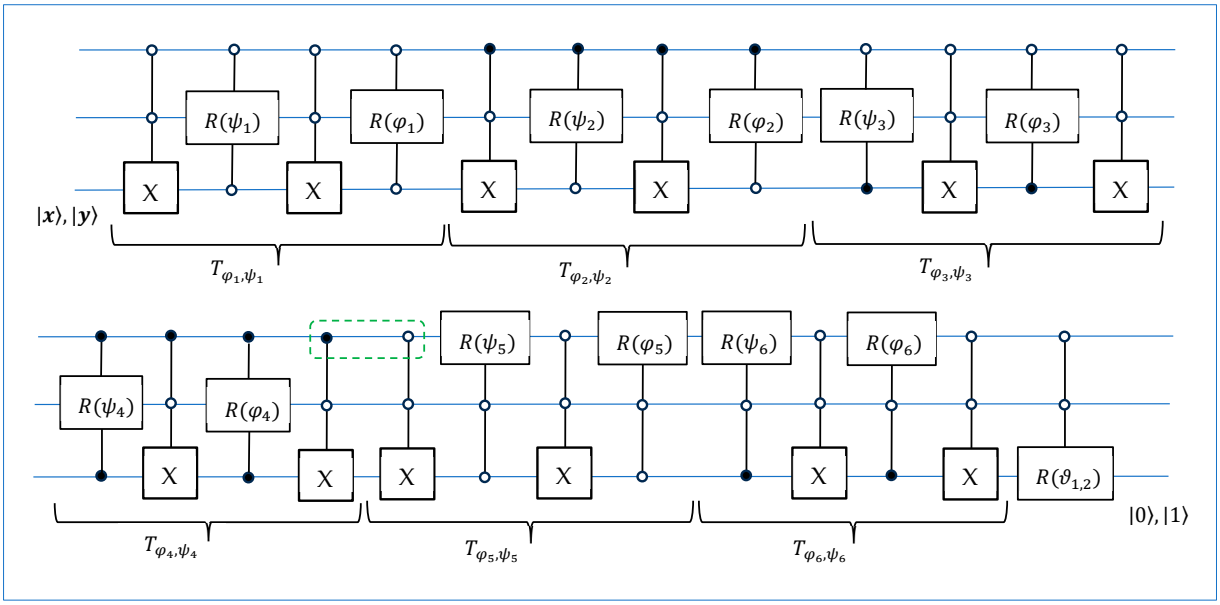


Figure 11. The quantum circuit for calculating the 3-qubit Q2siHT.

One can note that two controlled X gates connected in series at the junction of the operators T_{φ_4,ψ_4} and T_{φ_5,ψ_5} represent one X gate controlled by the second qubit. The circuit for calculating the inverse 3-qubit Q2siHT is given in Figure 12. The inverse rotations are denoted by $R'(\varphi) = R(-\varphi)$.

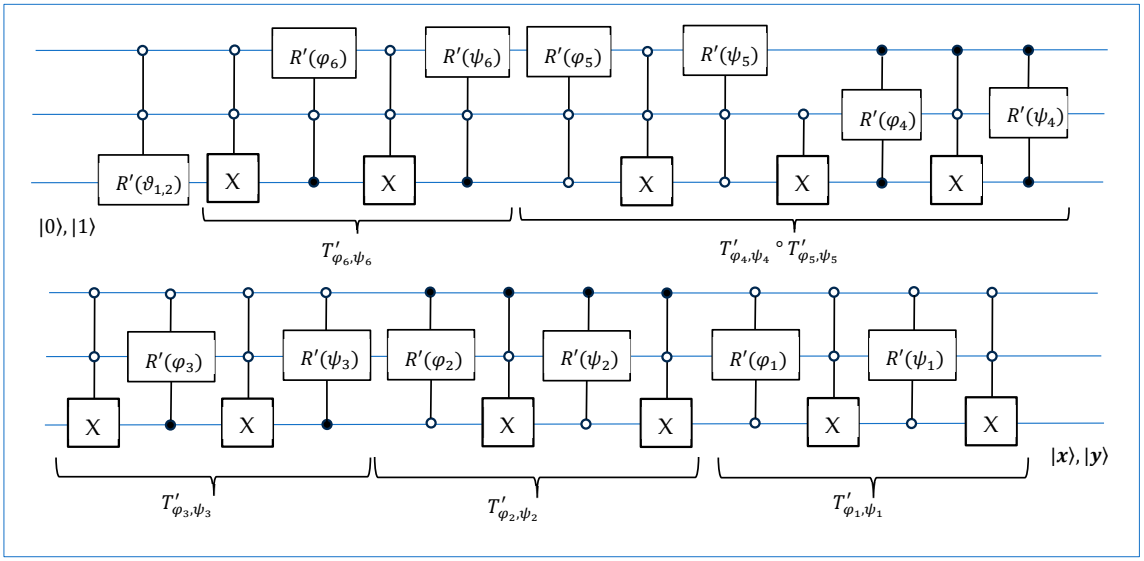


Figure 12. The quantum circuit for the preparation of two 3-qubit superpositions.

For comparison, we consider the traditional method $|000\rangle \to |\boldsymbol{x}\rangle$ of the preparation of the 3-qubit superposition by the Givens rotations. For this, we use the 8-point DsiHT with a fast path (for details, see [25]). The roadmap of this transformation of the 8D vector $\boldsymbol{x} \to (1, 0, \ldots, 0)$ is shown in Figure 13. Seven 2D rotations, or butterflies, are used, and all these rotations R_{ϑ_k}, $k = 1:7$ operate on the adjacent bit planes.

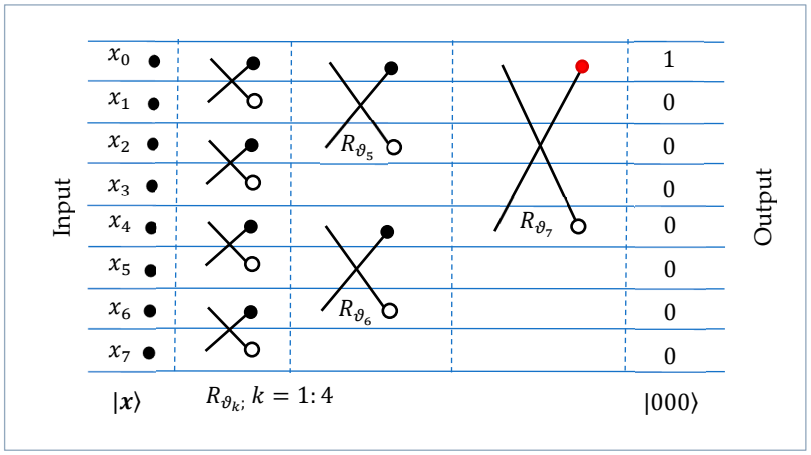


Figure 13. The roadmap of the 8-point DsiHT with a fast path.

The quantum circuit with seven 2-qubit controlled rotation gates for calculating this transformation, $|\boldsymbol{x}\rangle \to |000\rangle$, is given in Figure 14. This is a permutation-free circuit.

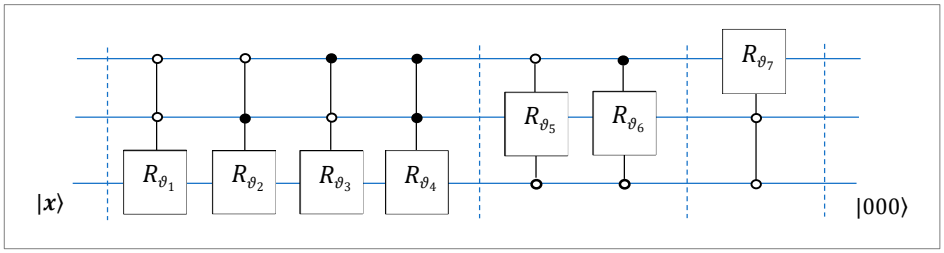


Figure 14. The circuit for the 3-qubit QsiHT with seven controlled rotation gates.

Figure 15 shows a roadmap of the inverse 8-point DsiHT. The transformation requires seven rotations with angles $-\vartheta_k$, $k = 1:7$. This transformation allows us to prepare the vector $\boldsymbol{x}$ when the input is the vector $(1, 0, \ldots, 0)$.

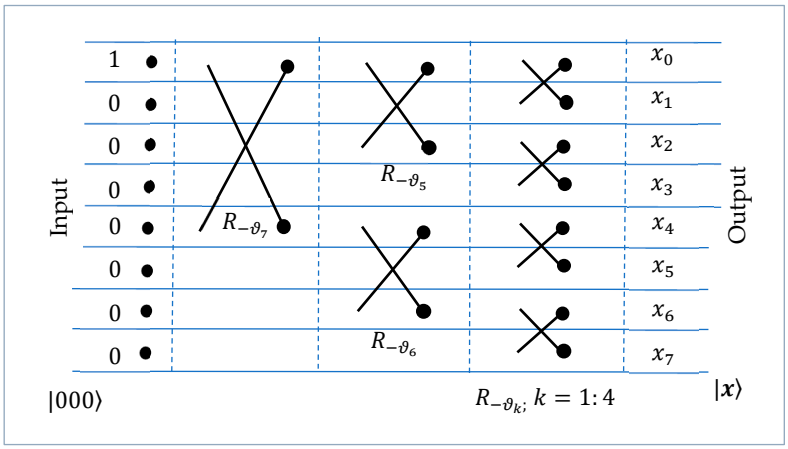


Figure 15. The roadmap of the 8-point inverse DsiHT with a fast path.

If we consider the generator $\boldsymbol{x} = (1, -2, 4, 5, -2, 5, 1, 3)'$, then the 8-point DsiHT is calculated by the rotations with the following angles:

$$A_{\boldsymbol{x}} = \{\vartheta_k, k = 1:7\} = \{63.4349^\circ, -51.3402^\circ, 68.1986^\circ, -71.5651^\circ, -70.7500^\circ, 30.4223^\circ, 42.6381^\circ\}. \quad (66)$$

The matrix of this transformation is equal to

$$H_{8;\mathrm{x}} = \begin{bmatrix} 0.1085 & -0.2169 & 0.4339 & 0.5423 & -0.2169 & 0.5423 & 0.1085 & 0.3254 \\ 0.8944 & 0.4472 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.4222 & 0.8444 & 0.2060 & 0.2574 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.7809 & 0.6247 & 0 & 0 & 0 & 0 \\ 0.0999 & -0.1997 & 0.3995 & 0.4994 & 0.2356 & -0.5890 & -0.1178 & -0.3534 \\ 0 & 0 & 0 & 0 & 0.9285 & 0.3714 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.1881 & -0.4702 & 0.2727 & 0.8181 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.9487 & 0.3162 \end{bmatrix}. \quad (67)$$

One can note that the first row of this matrix is the normalized vector $\boldsymbol{x}/\|\boldsymbol{x}\|$, as in the matrix H_8 for the 8-point D2siHT in Equation (43) (and in Equations (64) and (65)). The structure of this matrix (with 32 zeros) differs from that of the matrix of the 8-point weak DsiHT, which is shown in Equation (17) (and has only 21 zero coefficients). Therefore, the DsiHT with the fast path is considered more effective than the weak path DsiHT [26].

The 8-point DsiHT with the generator $\boldsymbol{y} = (2, 7, -6, 4, 1, -2, 5, 2)'$ has the following matrix (with the first row $\boldsymbol{y}/\|\boldsymbol{y}\|$):

$$H_{8;\mathrm{y}} = \begin{bmatrix} 0.1696 & 0.5937 & -0.5089 & 0.3393 & 0.0848 & -0.1696 & 0.4241 & 0.1696 \\ -0.9615 & 0.2747 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1933 & 0.6767 & 0.5911 & -0.3941 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5547 & 0.8321 & 0 & 0 & 0 & 0 \\ -0.0965 & -0.3379 & 0.2896 & -0.1931 & 0.1491 & -0.2981 & 0.7453 & 0.2981 \\ 0 & 0 & 0 & 0 & 0.8944 & 0.4472 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.4130 & 0.8260 & 0.3561 & 0.1424 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.3714 & 0.9285 \end{bmatrix}. \quad (68)$$

The angular representation of the generator is

$$A_{\boldsymbol{y}} = \{\vartheta_k, k = 1:7\} = \{-74.0546^\circ, 33.6901^\circ, 63.4349^\circ, -21.8014^\circ, 44.7272^\circ, -67.4504^\circ, -29.6417^\circ\}. \quad (69)$$

The quantum circuit for the inverse 3-qubit QsiHT with the roadmap in Figure 15 is shown in Figure 16. This circuit can be used for the preparation of any 3-qubit superposition $|\boldsymbol{z}\rangle$, by using the corresponding seven angles, $A_{\boldsymbol{z}}$. To prepare the 3-qubit superpositions $|\boldsymbol{x}\rangle$ and $|\boldsymbol{y}\rangle$, we need two separate circuits, that is, one circuit with angles of rotations of $A_{\boldsymbol{x}}$ for $|\boldsymbol{x}\rangle$ and another circuit with the angles of $A_{\boldsymbol{y}}$ for $|\boldsymbol{y}\rangle$. Each such circuit requires only seven 2-qubit controlled rotation gates.

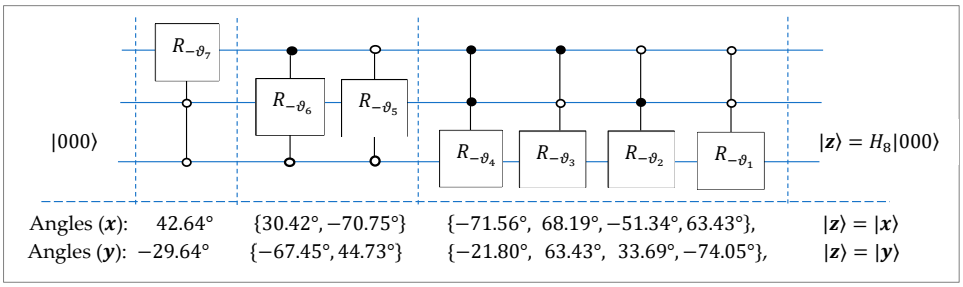


Figure 16. The circuits for the initiation of the 3-qubit state $|\boldsymbol{x}\rangle$.

5. Circuit Simulation and Analysis Using IBM's Qiskit Framework

The quantum circuits implementing the Q2siHT transform were simulated and analyzed using IBM's open-source Qiskit framework [29]. Qiskit is a framework for designing, optimizing, and executing quantum circuits. It includes realistic backends, various simulators, and visualization tools that enable direct comparison between ideal and noisy circuit executions within the same computational framework [30].

Two types of simulations were performed to evaluate the accuracy and noise sensitivity of the proposed circuits. The first simulation was conducted under noise-free (ideal) conditions to obtain the theoretical probability distributions of the output quantum states. The second simulation incorporated a noise model using Qiskit's GenericBackendV2, where the default noise parameters were applied. These parameters represent typical operational characteristics of IBM Quantum hardware such as gate readout errors and finite qubit coherence times. This allows for an approximate estimation of circuit behaviour under realistic experimental conditions [31].

To quantify the deviation between the simulated and theoretical results, the Root Mean Square Error (RMSE) was used as a performance metric, mathematically defined as

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1}\left(p_i^{\text{sim}} - p_i^{\text{ideal}}\right)^2}, \tag{70}$$

where p_i^{sim} and p_i^{ideal} denote the simulated and theoretical probabilities of the i-th quantum state, and N is the total number of possible output states. As shown in Equation (70), the RMSE provides a quantitative measure of the average deviation between the simulation results and theoretical expectations. A smaller RMSE indicates closer arrangement between simulated and theoretical distributions, which demonstrates the accuracy of the proposed Q2siHT-based quantum superposition preparation circuits under both ideal and noisy simulation environments.

The RMSE values and corresponding error distributions were obtained for the inverse Q2siHT fast path circuits of both the 2-qubit and 3-qubit systems. For each configuration, the results were generated under noise-free and noisy conditions. The numerical RMSE results are summarized in Tables 7–10, and their corresponding visual representations of the error distributions are shown in Figures 17–20.

Table 7. RMSE values for the inverse 2-qubit Q2siHT fast path circuit under noise-free simulation for $|x\rangle$ and $|y\rangle$.

Basis States	**Probabilities of $\|x\rangle$**				
	Theoretical	**500 Shots**	**1000 Shots**	**10,000 Shots**	**100,000 Shots**
00	0.1333	0.1160	0.1430	0.1369	0.1320
01	0.3000	0.3040	0.2980	0.3008	0.3007
10	0.0333	0.0320	0.0400	0.0326	0.0327
11	0.5333	0.5480	0.5190	0.5297	0.5346
RMSE		1.35×10^{-2}	1.20×10^{-2}	2.91×10^{-3}	1.40×10^{-3}
Basis States	**Probabilities of $\|y\rangle$**				
	Theoretical	**500 Shots**	**1000 Shots**	**10,000 Shots**	**100,000 Shots**
00	0.0217	0.0240	0.02400	0.0220	0.0221
01	0.3478	0.3340	0.3570	0.3430	0.3462
10	0.5435	0.5700	0.5300	0.5478	0.5459
11	0.0869	0.0720	0.0890	0.0872	0.0858
RMSE		1.74×10^{-2}	7.30×10^{-3}	2.56×10^{-3}	1.56×10^{-3}

Table 8. RMSE values for the inverse 2-qubit Q2siHT fast path circuit under noisy simulation (GenericBackendV2) for $|x\rangle$ and $|y\rangle$.

Basis States	**Probabilities of $\|x\rangle$**				
	Theoretical	**500 Shots**	**1000 Shots**	**10,000 Shots**	**100,000 Shots**
00	0.1333	0.1540	0.1380	0.1356	0.1349
01	0.3000	0.2680	0.3000	0.3057	0.2998
10	0.0333	0.0340	0.0450	0.0380	0.0382
11	0.5333	0.5440	0.5170	0.5207	0.5271
RMSE		2.06×10^{-2}	1.61×10^{-2}	8.12×10^{-3}	6.89×10^{-3}
Basis States	**Probabilities of $\|y\rangle$**				
	Theoretical	**500 Shots**	**1000 Shots**	**10,000 Shots**	**100,000 Shots**
00	0.0217	0.0300	0.0230	0.0210	0.0227
01	0.3478	0.3140	0.3370	0.3379	0.3478
10	0.5435	0.5740	0.5460	0.5494	0.5429
11	0.0869	0.0820	0.0940	0.0917	0.0866
RMSE		2.24×10^{-2}	7.83×10^{-3}	6.30×10^{-3}	1.68×10^{-3}

Table 9. RMSE values for the inverse 3-qubit Q2siHT fast path circuit under noise-free simulation for $|x\rangle$ and $|y\rangle$.

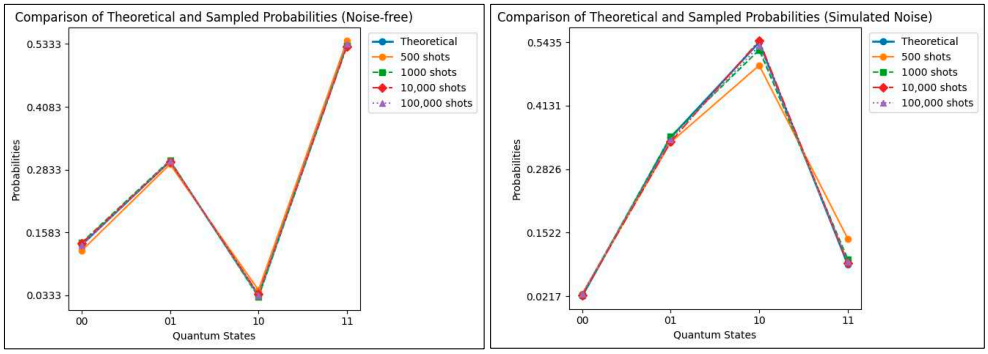
Basis States	**Probabilities of $\|x\rangle$**				
	Theoretical	**1000 Shots**	**10,000 Shots**	**100,000 Shots**	**1,000,000 Shots**
000	0.0118	0.0070	0.0088	0.0118	0.0117
001	0.0471	0.0420	0.0464	0.0466	0.0472
...	...	...	...	...	...
110	0.0118	0.0110	0.0105	0.0115	0.0119
111	0.1059	0.0910	0.1032	0.1070	0.106
RMSE		1.60×10^{-2}	6.63×10^{-3}	1.64×10^{-3}	3.81×10^{-4}
Basis States	**Probabilities of $\|y\rangle$**				
	Theoretical	**1000 Shots**	**10,000 Shots**	**100,000 Shots**	**1,000,000 Shots**
000	0.0288	0.0320	0.0303	0.0290	0.0287
001	0.3525	0.3330	0.3477	0.3527	0.3516
...	...	...	...	...	...
110	0.1799	0.1900	0.1771	0.1805	0.1800
111	0.0288	0.0290	0.0288	0.0284	0.0291
RMSE		9.87×10^{-3}	3.17×10^{-3}	1.55×10^{-3}	4.83×10^{-4}

Table 10. RMSE values for the inverse 3-qubit Q2siHT fast path circuit under noisy simulation (GenericBackendV2) for $|x\rangle$ and $|y\rangle$.

Basis States	**Probabilities of $\|x\rangle$**				
	Theoretical	**1000 Shots**	**10,000 Shots**	**100,000 Shots**	**1,000,000 Shots**
000	0.0118	0.066	0.0616	0.0623	0.0613
001	0.0471	0.101	0.0968	0.0985	0.0986
...	...	...	...	...	...
110	0.0118	0.068	0.0659	0.0673	0.0664
111	0.1059	0.1	0.1054	0.109	0.1091
RMSE		9.89×10^{-2}	9.78×10^{-2}	9.53×10^{-2}	9.42×10^{-2}

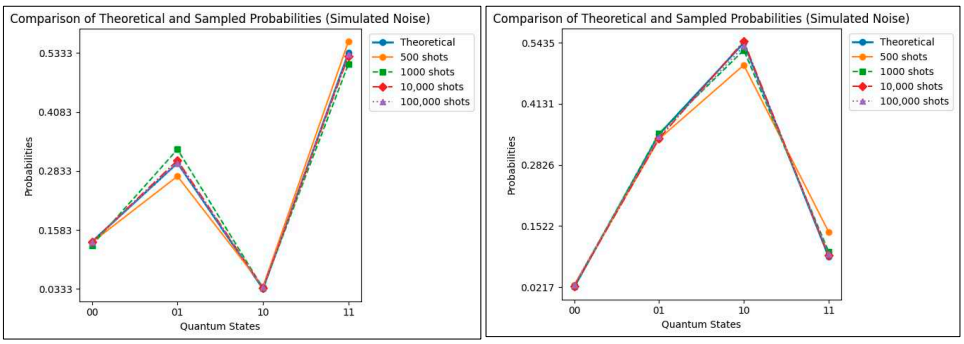
Table 10. *Cont.*

Basis States	Probabilities of $\|y\rangle$				
	Theoretical	**1000 Shots**	**10,000 Shots**	**100,000 Shots**	**1,000,000 Shots**
000	0.0288	0.0610	0.0688	0.0699	0.0716
001	0.3525	0.2380	0.2264	0.2311	0.2313
...	...	...	...	...	...
110	0.1799	0.1440	0.1527	0.1518	0.1494
111	0.0288	0.0900	0.0750	0.0752	0.0752
RMSE		1.07×10^{-1}	1.05×10^{-1}	1.02×10^{-1}	1.01×10^{-1}



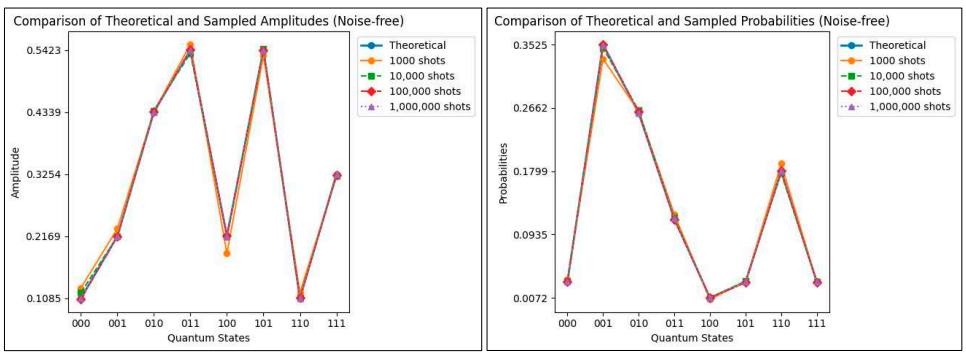
(**a**) (**b**)

Figure 17. RMSE distribution for the inverse 2-qubit Q2siHT fast path circuit for (**a**) $|x\rangle$ and (**b**) $|y\rangle$ (noise-free case).



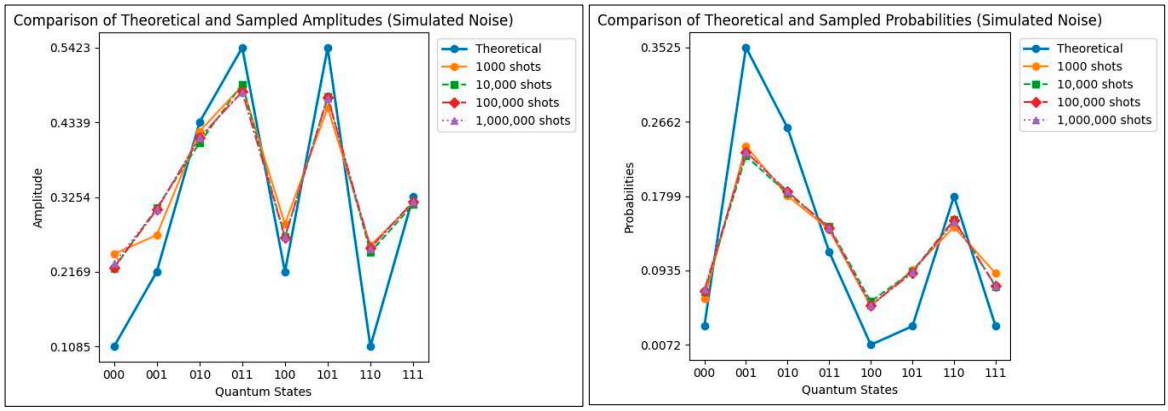
(**a**) (**b**)

Figure 18. RMSE distribution for the inverse 2-qubit Q2siHT fast path circuit for (**a**) $|x\rangle$ and (**b**) $|y\rangle$ (with noise).



(**a**) (**b**)

Figure 19. RMSE distribution for the inverse 3-qubit Q2siHT fast path circuit for (**a**) $|x\rangle$ and (**b**) $|y\rangle$ (noise-free case).



(**a**) (**b**)

Figure 20. RMSE distribution for the inverse 3-qubit Q2siHT fast path circuit for (**a**) $|x\rangle$ and (**b**) $|y\rangle$ (with noise).

It should be noted that the numerical examples presented in this work are used solely to demonstrate the operation and implementation of the proposed method and do not represent any specific physical or practical quantum systems. Furthermore, the described Q2siHT-based approach requires solving the decision equations to determine the rotation parameters. In this work, a classical computer is used to solve such equations.

When implementing these circuits through IBM's Qiskit framework, the single and controlled qubit rotation gates are represented in IEEE-754 double-precision format, which provides around 15–16 significant digits of numerical accuracy. In our simulations, all rotation parameters were maintained with a precision of 0.01°. It is important to note that the Qiskit transpiler may omit extremely small rotations (less than 10^{-8} rad) during optimization, so to prevent such removals, each computed rotation was preserved with at least four significant digits in degrees.

6. Conclusions

In this work, we present, for the first time, the discrete two signal-induced heap transformation (D2sHT). This transformation is generated by two given vectors or superpositions. The quantum analogue of this transform, namely the Q2siHT, is described, and the quantum circuits of its implementation are considered. We propose a new method of the preparation of the two states with 3D rotations by using the concept of the D2siHT. The amplitudes of quantum superpositions are considered real. Examples for 2- and 3-qubit superpositions are described with quantum circuits. It is also shown that this transformation can be implemented by using different paths of processing generators and input signals. Regarding the examples with the 3-qubit Q2siHT with strong and fast paths, the importance of this characteristic for the transformation is shown. Simulations using the Qiskit framework using the inverse Q2siHT fast path state preparation approach demonstrate that the Q2siHT is reliable and efficient for quantum state preparation. The low RMSE values in the noise-free simulations verify the theoretical precision of the proposed circuits, while the moderate error observed under noisy conditions demonstrates convergence and resilience to common sources of quantum decoherence. These results highlight their practical feasibility and application in future implementations on real quantum hardware. Although this work focuses primarily on the simultaneous preparation of two states in the same quantum circuit, the method of preparing individual states is demonstrated as well in Figures 15 and 16. This is performed to highlight that either quantum algorithm can be easily implemented for various design constraints where circuit fidelity may need to be minimized but certain gate parameters can be changed. In the case where only

one gate parameter can be changed, the Q2siHT is able to "multiplex", or switch between, two quantum superpositions by only changing the angle of one rotation gate.

The problem of two state preparation with complex amplitudes can also be considered and solved by the complex Q2siHT with three decision equations and complex rotation operations [25,32]. This case requires careful study. We need to initiate two superpositions $\boldsymbol{x}$ and $\boldsymbol{y}$. The real basic transformations $T_{\varphi,\psi} = R_\varphi R_\psi$ defined in Equation (22) are defined by the angles which are calculated from three 3D vectors, as shown in Equation (24). We need to solve angular Equation (23) when all triplets in Equation (26) are complex. We do not yet have a simple analytical solution to this problem. One possible way to resolve this is to remove the phases from the generators first

$$\begin{aligned}\boldsymbol{x} &= \left(e^{i\lambda_0}\left|x_0\right|, e^{i\lambda_1}\left|x_1\right|, \ldots, e^{i\lambda_{N-1}}\left|x_{N-1}\right| \right) \rightarrow \hat{\boldsymbol{x}} = \left(\left|x_0\right|, \left|x_1\right|, \ldots, \left|x_{N-1}\right| \right),\\ \boldsymbol{y} &= \left(e^{i\gamma_0}\left|y_0\right|, e^{i\gamma_1}\left|y_1\right|, \ldots, e^{i\gamma_{N-1}}\left|y_{N-1}\right| \right) \rightarrow \hat{\boldsymbol{y}} = \left(\left|y_0\right|, \left|y_1\right|, \ldots, \left|y_{N-1}\right| \right),\end{aligned}$$

and then initiate the superpositions $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ by the Q2siHTs described above. After that, we need to add the phase gates in the described circuits to restore the original superpositions $\boldsymbol{x}$ and $\boldsymbol{y}$. We will consider the solution of this problem in our future work.

Finally, although the Q2siHT is presented in an abstract algorithmic framework, its implementation in experimental quantum platforms with coherent control and multi-qubit superpositions may be achieved. For instance, cavity quantum electrodynamics systems provide the precise control of atom photon interactions and have been used to realize complex superpositions of atomic states [33]. Similarly, superconducting qubits and trapped ions enable scalable and high-fidelity multi-qubit operations that could potentially serve as realistic environments for testing the proposed Q2siHT circuits. In these experimental platforms, the controlled rotations and operations required by the Q2siHT could be directly available as gate primitives, which would allow the proposed circuits to be benchmarked using current hardware to evaluate their resilience to noise, decoherence, and gate depth constraints [34].

Author Contributions: Conceptualization, A.M.G.; methodology, A.M.G.; software, A.M.G. and A.A.G.; validation, A.M.G. and A.A.G.; formal analysis, A.M.G. and A.A.G.; investigation, A.M.G.; resources, A.M.G.; data curation, A.M.G. and A.A.G.; writing—original draft preparation, A.M.G.; writing—review and editing, A.M.G. and A.A.G.; visualization, A.M.G. and A.A.G.; supervision, A.M.G.; project administration, A.M.G.; funding acquisition, A.M.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DsiHT	Discrete signal-induced heap transform
D2siHT	Discrete two signal-induced heap transform
Q2siHT	Quantum two signal-induced heap transform
BP	Bit plane

References

1. Deutsch, D. Quantum computational networks. *Proc. R. Soc. Lond. A Math. Phys. Sci.* **1989**, *425*, 73–90.
2. Deutsch, D.; Barenco, A.; Ekert, A. Universality in quantum computation. *Proc. R. Soc. Lond. A Math. Phys. Sci.* **1995**, *449*, 669–677.
3. Barenco, A.; Bennett, C.H.; Cleve, R.; DiVincenzo, D.P.; Margolus, N.; Shor, P.; Sleator, T.; Smolin, J.A.; Weinfurter, H. Elementary gates for quantum computation. *Phys. Rev. A* **1995**, *52*, 3457. [CrossRef] [PubMed]
4. Knill, E. *Approximation by Quantum Circuits*; LANL Rep. LAUR-95-2225; Los Alamos National Laboratory: Los Alamos, NM, USA, 1995.
5. Long, G.-L.; Sun, Y. Efficient scheme for initializing a quantum register with an arbitrary superposed state. *Phys. Rev. A* **2001**, *64*, 014303. [CrossRef]
6. Shende, V.V.; Markov, I.L. Quantum circuits for incompletely specified two-qubit operators. *Quantum Inf. Comput.* **2005**, *5*, 49–58. [CrossRef]
7. Bärtschi, A.; Eidenbenz, S. Deterministic preparation of Dicke states. In *International Symposium on Fundamentals of Computation Theory*; Springer: Cham, Switzerland, 2019.
8. Plesch, M.; Brukner, C. Quantum state preparation with universal gate decompositions. *arXiv* **2010**, arXiv:1003.5760v2. [CrossRef]
9. Gleinig, N.; Hoefler, T. An efficient algorithm for sparse quantum state preparation. In Proceedings of the 58th ACM/IEEE Design Automation Conference, San Francisco, CA, USA, 5–9 December 2021; IEEE: New York, NY, USA, 2021.
10. de Veras, T.M.; da Silva, L.D.; da Silva, A.J. Double sparse quantum state preparation. *Quantum Inf. Process.* **2002**, *21*, 204. [CrossRef]
11. Volya, D.; Mishra, P. State preparation on quantum computers via quantum steering. *arXiv* **2023**, arXiv:2302.13518v3. [CrossRef]
12. Householder, A.S. Unitary Triangulation of a Nonsymmetric Matrix. *J. ACM* **1958**, *5*, 339–342. [CrossRef]
13. Rutishause, H. Simultaneous Iteration Method for Symmetric Matrices. In *Linear Algebra. Handbook for Automatic Computation*; Bauer, F.L., Ed.; Springer: Berlin/Heidelberg, Germany, 2017.
14. Golub, G.H.; Van Loan, C.F. *Matrix Computations*, 3rd ed.; Johns Hopkins Press: Baltimore, ML, USA, 1996.
15. Bindel, D.; Demmel, J.; Kahan, W.; Marques, O. *On Computing Givens Rotations Reliably and Efficiently*; LAPACK Working Note 148; UT-CS-00-449; University of Tennessee: Knoxville, TN, USA, 2001.
16. Demmel, J.; Grigori, L.; Hoemmen, M.; Langou, J. Communication-optimal parallel and sequential QR and LU factorizations. *SIAM J. Sci. Comp.* **2012**, *34*, 206–239. [CrossRef]
17. Wang, H.; Bochen, T.D.; Cong, J. Quantum state preparation circuit optimization exploiting don't cares. *arXiv* **2024**, arXiv:2409.01418.
18. Pinto, D.F.; Friedrich, L.; Maziero, J. Preparing general mixed quantum states on quantum computers. *arXiv* **2024**, arXiv:2402.04212. [CrossRef]
19. Zhang, X.M.; Li, T.; Yuan, X. Quantum state preparation with optimal circuit depth: Implementations and applications. *Phys. Rev. Lett.* **2022**, *129*, 230504. [CrossRef] [PubMed]
20. Vartiainen, J.J.; Möttönen, M.; Salomaa, M.M. Efficient decomposition of quantum gates. *Phys. Rev. Lett.* **2004**, *92*, 177902. [CrossRef] [PubMed]
21. Möttönen, M.; Vartiainen, J.J.; Bergholm, V.; Salomaa, M.M. Quantum circuits for general multiqubit gates. *Phys. Rev. Lett.* **2004**, *93*, 130502. [CrossRef]
22. Bergholm, V.; Vartiainen, J.J.; Möttönen, M.; Salomaa, M.M. Quantum circuits with uniformly controlled one-qubit gates. *arXiv* **2004**, arXiv:quant-ph/0410066v2. [CrossRef]
23. Shende, V.V.; Bullock, S.S.; Markov, I.L. Synthesis of quantum-logic circuits. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **2006**, *25*, 1000–1010. [CrossRef]
24. Grigoryan, A.M.; Gomez, A.; Agaian, S.S. A Novel approach to state-to-state transformation in quantum computing. *Information* **2025**, *16*, 689. [CrossRef]
25. Grigoryan, A.M. New permutation-free quantum circuits for implementing 3- and 4-qubit unitary operations. *Information* **2025**, *16*, 621. [CrossRef]
26. Grigoryan, A.M.; Grigoryan, M.M. *Brief Notes in Advanced DSP: Fourier Analysis with MATLAB*; CRC Press: Boca Raton, FL, USA; Taylor, and Francis Group: Abingdon, UK, 2009.
27. Nielsen, M.A.; Chuang, I.L. *Quantum Computation and Quantum Information*; Cambridge University Press: Cambridge, UK, 2000.
28. Rieffel, E.G.; Polak, W.H. *Quantum Computing: A Gentle Introduction*; The MIT Press: Cambridge, MA, USA, 2011.
29. Qiskit Development Team. *Qiskit: An Open-Source Framework for Quantum Computing*, Version 1.3.2. Computer Software. IBM Quantum: Poughkeepsie, NY, USA, 2019.
30. Mykhailova, M. *Quantum Programming in Depth Solving problems with Q# and Qiskit*; Manning: Shelt Island, NY, USA, 2025.
31. IBM Quantum. Exact and Noisy Simulation with Qiskit Aer Primitives. IBM Quantum Documentation—Guides, 2024–2025. Available online: https://quantum.cloud.ibm.com/docs/guides/simulate-with-qiskit-aer (accessed on 1 November 2025).
32. Grigoryan, A.M. Effective methods of QR-decompositions of square complex matrices by fast discrete signal-induced heaptransforms. *Adv. Linear Algebra Matrix Theory (ALAMT)* **2023**, *12*, 87–110. [CrossRef]

33. Haroche, S.; Raimond, J.-M. *Exploring the Quantum: Atoms, Cavities, and Photons*; Oxford University Press: Oxford, UK, 2006; ISBN 0-19-850914-6.
34. Häffner, H.; Roos, C.F.; Blatt, R. Quantum Computing with Trapped Ions. *Phys. Rep.* **2008**, *469*, 155–203. [CrossRef]