

An Implementation Method of Digital FIR Filter Using Quantum State Preparation Circuit

Chien-Cheng Tseng

Department of Computer and Communication Engineering
National Kaohsiung University of Science and Technology
Kaohsiung, Taiwan
E-mail: tcc@nku.edu.tw

Su-Ling Lee

Department of Computer Science and Information Engineering
Chang Jung Christian University
Tainan, Taiwan
E-mail: lilee@mail.cjcu.edu.tw

Abstract—Digital finite-impulse-response (FIR) filters are a cornerstone of digital signal processing (DSP) and modern communication systems, making their implementation via quantum computation a compelling area of study. Although recent methods have used orthogonal triangular decomposition (OTD) to realize FIR filters through quantum elementary gates, these approaches rely heavily on matrix QR decomposition and IBM Qiskit's transpilation tools, resulting in significant computational overhead. To address these inefficiencies, this paper proposes a new implementation using quantum state preparation (QSP) circuits. By employing only rotation y gate and multi-controlled rotation y gates, the proposed method determines circuit parameters through closed-form solutions derived directly from the input signal and filter coefficient vectors. Experimental results for a digital mean filter demonstrate the effectiveness of this QSP-based approach, showing the reduced implementation complexity compared to the conventional OTD method.

Keywords—quantum computation, state preparation circuit, elementary gate, digital filter, FIR filter

I. INTRODUCTION

A digital filter is a mathematical algorithm or processor that operates on discrete-time signals to suppress noise, enhance frequency components, or perform signal shaping [1]. Unlike analog counterparts—which rely on physical components like resistors and capacitors—digital filters leverage digital signal processing (DSP) to manipulate numerical sequences representing real-world data [2]. They are broadly categorized into finite impulse response (FIR) filters, known for their inherent stability and linear phase, and infinite impulse response (IIR) filters, which offer greater computational efficiency through feedback loops. By utilizing techniques such as convolution or difference equations, these filters provide precise control over signal characteristics that are often unattainable in the analog domain. Today, they are ubiquitous, powering everything from high-fidelity audio to noise-canceling headphones. In modern communication, digital filters manage the crowded electromagnetic spectrum by performing critical functions like pulse shaping to prevent spectral bleeding and channel equalization to correct transmission distortions. By implementing sharp cut-off frequencies and linear phase responses, they minimize inter-symbol interference (ISI) to ensure high-speed data integrity. Furthermore, their capacity for sophisticated noise reduction and echo cancellation enables clear communication in high-interference environments, remaining the fundamental enabler of global, high-speed connectivity as modulation schemes grow increasingly complex.

On the other hand, quantum computation represents a paradigm shift from classical processing by leveraging the principles of quantum mechanics—specifically superposition and entanglement—to manipulate information in ways previously deemed impossible [3]. While a classical bit is

restricted to a binary state of 0 or 1, a quantum bit, or qubit, can exist in a linear combination of both states simultaneously. This allows a quantum system of m qubits to represent 2^m states at once, providing massive, inherent parallelism. The true hallmark of its computational efficiency, however, is not merely performing simultaneous operations, but rather utilizing quantum interference to amplify the probability of the correct output while canceling out incorrect ones. For specific applications, this leads to dramatic speedups; for example, Shor's algorithm provides an exponential acceleration for integer factorization compared to the best-known classical algorithms, while Grover's algorithm offers a quadratic speedup for unstructured database searches. These efficiencies have empowered researchers to address complex challenges in digital image processing, detection and estimation, artificial intelligence, combinatorial optimization, data clustering and machine learning [4]. Driven by these advancements, this paper aims to investigate the implementation of a digital FIR filter using a quantum state preparation circuit.

While orthogonal triangular decomposition (OTD) has recently been utilized to implement FIR filters via quantum elementary gates, the approach relies heavily on matrix QR decomposition and IBM Qiskit's transpilation tools, incurring substantial computational overhead [5]. To mitigate these inefficiencies, this paper proposes a streamlined implementation utilizing quantum state preparation (QSP) circuits [6]. By employing only rotation y gate and multi-controlled rotation y gates, the proposed method determines circuit parameters through closed-form solutions derived directly from the input signal and filter coefficient vectors, ensuring a more accessible and user-friendly framework. The remainder of this paper is organized as follows: Section II introduces the fundamentals of quantum computation and the design of QSP circuits. Section III defines the quantum FIR filter implementation problem, reviews the conventional OTD method, and details the proposed QSP-based approach. Section IV presents experimental results for a digital mean filter that demonstrate the effectiveness of our QSP method through the reduced complexity of quantum gates. Finally, Section V concludes the paper with a discussion on future research directions.

II. PRELIMINARIES

In this section, the fundamentals of quantum computation are briefly described and a design method of quantum state preparation circuit is studied.

A. Basics of Quantum Computation

The basic unit of quantum computation is quantum bit (qubit). A quantum register is composed of m qubits working together to store and manipulate information. The state $|\psi\rangle$ of register can exist in a superposition of all 2^m possible values simultaneously, so it can be represented by

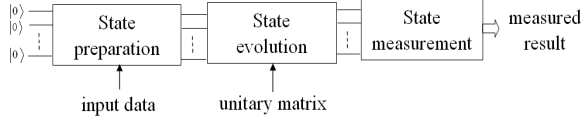


Fig.1 Three-step procedure of the quantum computation in quantum register.

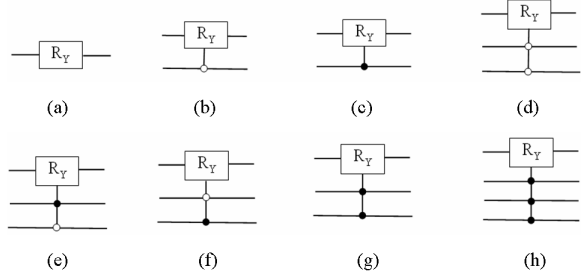


Fig.2 The quantum elementary gates of rotation y gate and controlled rotation y gates. (a) $R_y(\theta)$ gate. (b) $\bar{C}R_y(\theta)$ gate. (c) $CR_y(\theta)$ gate. (d) $\bar{C}\bar{C}R_y(\theta)$ gate. (e) $\bar{C}CR_y(\theta)$ gate. (f) $CCR_y(\theta)$ gate. (g) $CCR_y(\theta)$ gate. (h) $CCCR_y(\theta)$ gate.

$$|\psi\rangle = \sum_{k=0}^{2^m-1} \alpha_k |k\rangle \quad (1)$$

where $\alpha_0, \alpha_1, \dots$, and α_{2^m-1} are probability amplitudes and k is the decimal representation. When the register is measured, it collapses into one of basis states $|k\rangle$ with probability $|\alpha_k|^2$. Thus, the following probability rule needs to be satisfied:

$$\sum_{k=0}^{2^m-1} |\alpha_k|^2 = 1 \quad (2)$$

Because a quantum register can represent a massive range of values at once, a quantum algorithm can perform calculations on all those values simultaneously. This is known as quantum parallelism. As Fig.1 shown, the computation usually happens in register through the following three steps:

Step 1: State preparation: Set all qubits to a wanted state from the initial zero state $|0\rangle$.

Step 2: State evolution: Apply various unitary operations of elementary quantum gates that flip, rotate, or entangle the qubits within register to perform the wanted computation tasks.

Step 3: State measurement: When the computation is finished, the register is measured. At this point, the superposition collapses, and the register returns a single classical bit string of 0 and 1.

The quantum circuits to implement the above quantum state preparation and state evolution are usually constructed by using quantum elementary gates including single-qubit, double-qubit and triple-qubit gates. Fig.2(a)-(h) show the rotation y gate, two single-controlled rotation y gates, four double-controlled rotation y gates, and a triple-controlled rotation y gate, where θ is rotation angle of gate. Notation $\bar{C}$ denotes the white-dot control qubit and C denotes the black-dot control qubit. The unitary matrix $R_y(\theta)$ to describe the input/output (I/O) relation of rotation y gate in Fig.2(a) is given by

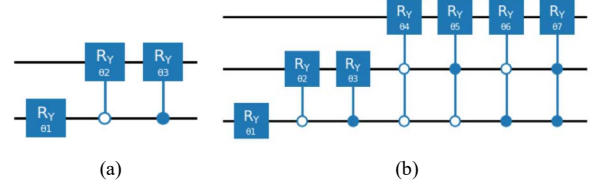


Fig.3 Two quantum state preparation (QSP) circuits. (a) The QSP circuit with $m=2$. (b) The QSP circuit with $m=3$.

$$R_y(\theta) = \begin{bmatrix} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{bmatrix} \quad (3)$$

The unitary matrices to describe the I/O relations of single-controlled rotation y gates in Fig.2(b)(c) are given by

$$\bar{C}R_y(\theta) = \begin{bmatrix} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} & 0 & 0 \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad CR_y(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \\ 0 & 0 & \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{bmatrix} \quad (4)$$

As to the unitary matrices $\bar{C}\bar{C}R_y(\theta)$, $\bar{C}CR_y(\theta)$, $CC\bar{C}R_y(\theta)$ and $CCCR_y(\theta)$ to describe I/O relations of double-controlled rotation y gates can be found in [3]. In next subsection, the details that use these rotation y gates to design state preparation circuit are described.

B. Quantum State Preparation Circuit

To process classical data via quantum computation, the input must first be encoded into a superposition state. This allows quantum basic gates to manipulate the state evolution, ultimately converging on the desired final state. For a given data vector $\alpha = [\alpha_0, \alpha_1, \dots, \alpha_{2^m-1}]^T$, this quantum state preparation problem is how to design a quantum circuit to transform the data vector α into the superposition state $|\psi\rangle$ in (1) from the zero state $|0\rangle$. That is, we seek a unitary operator U such that

$$U|0\rangle = |\psi\rangle = \sum_{k=0}^{2^m-1} \alpha_k |k\rangle \quad (5)$$

In [6], the rotation y gate and controlled rotation y gates in Fig.2 have been applied to design a circuit for implementing unitary operator U in (5). Two of these circuits are shown in Fig.3 for the cases of $m=2, 3$. The angle parameters in gates can be determined from the data vector α . For the case of $m=3$, seven parameters $\theta_1, \theta_2, \dots$, and θ_7 are given by

$$\begin{aligned} \theta_1 &= 2 \tan^{-1} \left(\sqrt{\frac{\alpha_4^2 + \alpha_5^2 + \alpha_6^2 + \alpha_7^2}{\alpha_0^2 + \alpha_1^2 + \alpha_2^2 + \alpha_3^2}} \right) \\ \theta_2 &= 2 \tan^{-1} \left(\sqrt{\frac{\alpha_2^2 + \alpha_3^2}{\alpha_0^2 + \alpha_1^2}} \right) \quad \theta_3 = 2 \tan^{-1} \left(\sqrt{\frac{\alpha_6^2 + \alpha_7^2}{\alpha_4^2 + \alpha_5^2}} \right) \\ \theta_4 &= 2 \tan^{-1} \left(\frac{\alpha_1}{\alpha_0} \right) \quad \theta_5 = 2 \tan^{-1} \left(\frac{\alpha_3}{\alpha_2} \right) \\ \theta_6 &= 2 \tan^{-1} \left(\frac{\alpha_5}{\alpha_4} \right) \quad \theta_7 = 2 \tan^{-1} \left(\frac{\alpha_7}{\alpha_6} \right) \end{aligned} \quad (6)$$

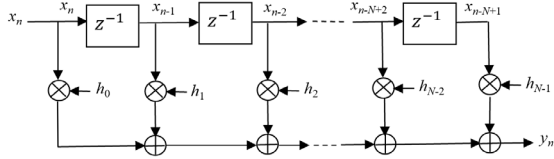


Fig.4 The direct-form implementation structure of FIR digital filter, where notation z^{-1} is the unit delay element.

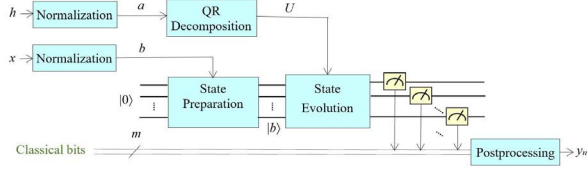


Fig.5 The block diagram of conventional QR decomposition method used to implement the FIR digital filter in quantum way.

Detailed steps for obtaining the closed-form formula in (6) are provided in [6]. We leverage existing QSP circuits to implement the digital FIR filter in the next section, as they are both efficient and user-friendly.

III. IMPLEMENTATION OF FIR FILTER USING QSP CIRCUIT

The input/output (I/O) relationship of FIR digital filter can be expressed as the time-domain difference equation of the input signal x_n and the filter coefficient h_k below:

$$y_n = \sum_{k=0}^{N-1} h_k x_{n-k} \quad (7)$$

where x_n is the current input sample, x_{n-k} are past input samples held in a tap delay line, h_k represents the feed-forward filter coefficients and N is the number of filter taps. Here, we assume N is a positive power-of-two integer, i.e., $N = 2^m$. Fig.4 shows a direct-form implementation structure of this filter which requires $N-1$ delay elements, $N-1$ adders, and N multipliers. The incoming signal x_n passes through a chain of delay elements, and each delayed sample x_{n-k} is multiplied by its corresponding coefficient h_k . Finally, all multiplied terms are summed together to produce the output signal y_n . Although the structure in Fig.4 can implement FIR filter successfully, it needs digital hardware and software. Due to rapid progress of quantum computing technology, this paper aims to implement FIR filter in quantum way which use quantum hardware and software. To achieve this purpose, let us define two vectors below:

$$h = [h_0 \ h_1 \ \dots \ h_{N-1}]^T \quad (8)$$

$$x = [x_n \ x_{n-1} \ \dots \ x_{n-N+1}]^T \quad (9)$$

where T denotes the vector or matrix transpose. Thus, the output signal y_n in (7) can be expressed as

$$y_n = h^T x \quad (10)$$

In [5], an orthogonal triangular decomposition which is also called QR decomposition has been presented to implement FIR filter in quantum way. The block diagram of this method is shown in Fig.5 and its procedure is listed below:

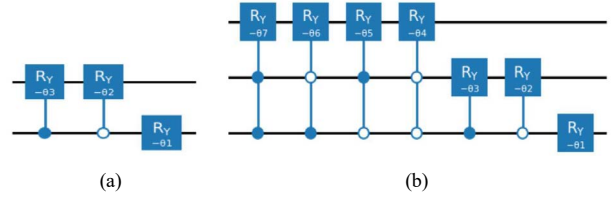


Fig.6 The inverse QSP circuit corresponding to inverse matrix $U^\dagger$. (a) The circuit with $m=2$. (b) The circuit with $m=3$.

Step 1: Due to the probability constraint in quantum register, the vectors h and x need to be normalized to unit-norm below:

$$a = \frac{h}{c_h} = \left[\frac{h_0}{c_h} \ \frac{h_1}{c_h} \ \dots \ \frac{h_{N-1}}{c_h} \right]^T = [a_0 \ a_1 \ \dots \ a_{N-1}]^T \quad (11)$$

$$b = \frac{x}{c_x} = \left[\frac{x_n}{c_x} \ \frac{x_{n-1}}{c_x} \ \dots \ \frac{x_{n-N+1}}{c_x} \right]^T = [b_0 \ b_1 \ \dots \ b_{N-1}]^T \quad (12)$$

where constants c_h and c_x are given by

$$c_h = \|h\| = \sqrt{\sum_{k=0}^{N-1} |h_k|^2} \quad c_x = \|x\| = \sqrt{\sum_{k=0}^{N-1} |x_{n-k}|^2} \quad (13)$$

Step 2: Use the QSP circuit in Fig.3 to map vector b into the following superposition state $|b\rangle = \sum_{k=0}^{N-1} b_k |k\rangle$.

Step 3: Replace the first column vector of identity matrix with size $N \times N$ to get an auxiliary matrix, then perform QR decomposition of this matrix to obtain the unitary matrix U .

Step 4: Use the IBM Qiskit's transpilation tools to get the quantum circuit of unitary matrix U and let state $|b\rangle$ pass through this circuit to perform unitary evolution for getting the state $|d\rangle = U|b\rangle$.

Step 5: Measure the state $|d\rangle$ many times to get its corresponding histogram, then readout the probability z of zero state from the histogram.

Step 6: In the post-processing task, the output signal y_n is computed as $y_n = c_h c_x \sqrt{z}$.

Although the above conventional method can be used to implement FIR filter in quantum way, this approach requires to perform QR decomposition and IBM Qiskit's transpilation tools, resulting in significant computation road. Thus, this paper aims to remove this two drawbacks. To achieve this purpose, let us study inverse of quantum state preparation (QSP) circuit in Fig.3. If the QSP circuit is represented by a unitary matrix U , the inverse is its conjugate transpose, denoted as $U^\dagger$ where the superscript $\dagger$ is dagger notation. Because quantum gates are unitary, the inverse circuit can be found by following two primary rules: reverse the order of the gates and invert each individual gate. As an example, the matrix U of the circuit in Fig.3(a) with $m = 2$ is given by

$$U = CR_y(\theta_3) \bar{C}R_y(\theta_2) (R_y(\theta_1) \otimes I) \quad (14)$$

where $\otimes$ is tensor product and I is the identity matrix. Thus, the inverse matrix $U^\dagger$ can be written as

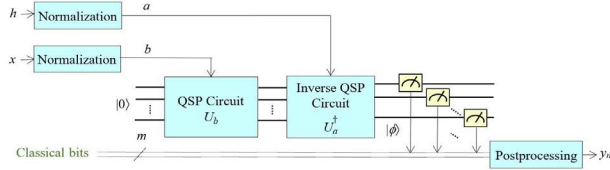


Fig.7 The block diagram of the proposed QSP circuit method used to implement the FIR digital filter in quantum way.

$$U^\dagger = (R_y(\theta_1) \otimes I)^\dagger \bar{C}R_y(\theta_2)^\dagger CR_y(\theta_3)^\dagger \quad (15)$$

Because rotation y gate is defined by a specific angle, its inversion is simply the gate by changing the sign of the angle. Therefore, $U^\dagger$ in (15) can be rewritten as

$$U^\dagger = (R_y(-\theta_1) \otimes I) \bar{C}R_y(-\theta_2) CR_y(-\theta_3) \quad (16)$$

Based on the above results, the inverse QSP circuit with $m = 2$ is shown in Fig.6(a). As to the circuit in $m = 3$ case, it can be obtained by the similar way and is shown in Fig.6(b). In the following, let us study how to use the QSP circuits in Fig.3 and Fig.6 to implement FIR filter. Substituting (11) and (12) into (10), we have

$$y_n = h^T x = c_h c_x a^T b \quad (17)$$

From the superposition states $|a\rangle = \sum_{k=0}^{N-1} a_k |k\rangle$ and $|b\rangle = \sum_{k=0}^{N-1} b_k |k\rangle$, we have the relation $a^T b = \langle a|b\rangle$. Thus, (17) reduces to

$$y_n = c_h c_x \langle a|b\rangle \quad (18)$$

So, the problem reduces to efficiently compute inner product $\langle a|b\rangle$ in quantum way. Using QSP circuit in Fig.3, the states $|a\rangle$ and $|b\rangle$ can be written as forms

$$|a\rangle = U_a |0\rangle \quad |b\rangle = U_b |0\rangle \quad (19)$$

where U_a and U_b are unitary matrices of the QSP circuits to prepare states $|a\rangle$ and $|b\rangle$. Therefore, the inner product $\langle a|b\rangle$ can be computed by

$$\langle a|b\rangle = |a\rangle^\dagger |b\rangle = (U_a |0\rangle)^\dagger U_b |0\rangle = \langle 0|U_a^\dagger U_b |0\rangle \quad (20)$$

Let us define a state $|\phi\rangle$ as follows:

$$|\phi\rangle = U_a^\dagger U_b |0\rangle \quad (21)$$

then (20) can be rewritten as $\langle a|b\rangle = \langle 0|\phi\rangle$. If the state $|\phi\rangle$ is expressed as $|\phi\rangle = \sum_{k=0}^{N-1} \beta_k |k\rangle$, we have the result

$$\langle 0|\phi\rangle = \sum_{k=0}^{N-1} \beta_k \langle 0|k\rangle = \beta_0 \quad (22)$$

This is because the following orthogonal property holds:

$$\langle 0|k\rangle = \begin{cases} 1 & k = 0 \\ 0 & k = 1, 2, \dots, N-1 \end{cases} \quad (23)$$

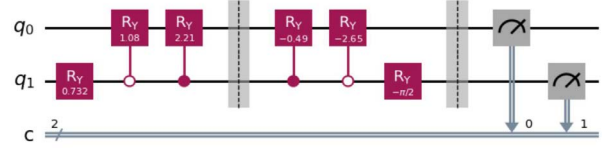


Fig.8 The quantum circuit of the proposed QSP circuit method for $N=4$ ($m=2$) case. The normalization and postprocessing are not shown here.

When the state $|\phi\rangle$ is measured, the probability p_0 to get the zero state $|0\rangle$ is given by $p_0 = |\beta_0|^2$. Combining the above results, it yields

$$\langle a|b\rangle = \langle 0|\phi\rangle = \beta_0 = \sqrt{p_0} \quad (24)$$

if β_0 is a real non-negative number. Substituting (24) into (18), the output signal y_n is given by

$$y_n = c_h c_x \sqrt{p_0} \quad (25)$$

From the above derivations, the block diagram to compute output signal y_n using QSP circuits is shown in Fig.7. Three tasks of this method are described below:

Task 1: Use the normalization in (11) and (12) to compute vectors a and b from the vectors h and x . Then, the angle parameters of QSP circuits are calculated and loaded into the quantum circuits of $U_a^\dagger$ and U_b . Finally, the state $|\phi\rangle$ is obtained by using the cascade circuits of U_b and $U_a^\dagger$, that is, $|\phi\rangle = U_a^\dagger U_b |0\rangle$.

Task 2: Measure the superposition state $|\phi\rangle$ to get its probability distribution or histogram. Then, the zero-state probability p_0 can be obtained from the histogram.

Task 3: In this post-processing task, the output signal y_n is computed as $y_n = c_h c_x \sqrt{p_0}$.

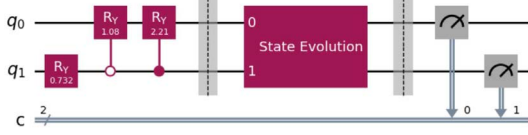
From Fig.7, it can be seen that the proposed method only uses the QSP circuit U_b and inverse QSP circuit $U_a^\dagger$. Thus, QR decomposition and IBM Qiskit's transpilation tools are not needed in the proposed method. In next section, experiment results are presented to show that implementation complexity of quantum gates can be reduced by QSP method when it is compared with QR decomposition method.

IV. EXPERIMENTAL RESULTS

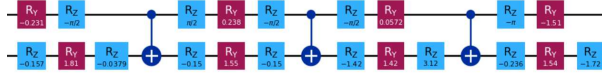
In this section, digital mean filters with $N=4$ ($m=2$) and $N=8$ ($m=3$) are used to evaluate performance. The proposed QSP method is also compared with the conventional QR decomposition method to show its effectiveness. The input signal x_n is selected as the following sequence: $x_0 = 2$, $x_1 = 1$, $x_2 = 3$, $x_3 = 5$, $x_4 = 4$, $x_5 = 1$, $x_6 = 2$, $x_7 = 6$, $x_8 = 3$. Now, two examples using this signal x_n are illustrated below:

Example 1: In this example, the case of $N = 4$ ($m = 2$) is studied. The filter coefficient vector h of mean filter is chosen as

$$h = [h_0 \ h_1 \ h_2 \ h_3]^T = [0.1 \ 0.4 \ 0.4 \ 0.1]^T \quad (26)$$



(a)



(b)

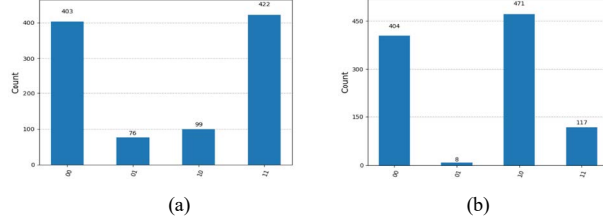


Fig.10 The histograms of the measured results when quantum circuits are executed 1000 times. (a) The histogram of the proposed QSP circuit method. (b) The histogram of the conventional QR decomposition method.

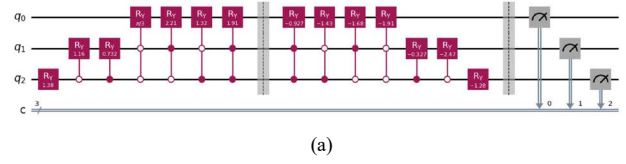
Without losing generality, the computation of output signal y_3 is considered here, so the signal vector x is given by

$$x = [x_3 \ x_2 \ x_1 \ x_0]^T = [5 \ 3 \ 1 \ 2]^T \quad (27)$$

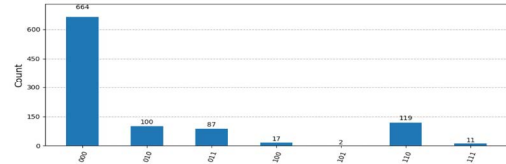
Traditional digital approach gives the output signal as $y_3 = h^T x = 2.3$. In the following, let us show the results of quantum computation methods. The constants for signal and coefficient normalization in (11) and (12) are given by $c_h = 0.583$ and $c_x = 6.245$. So, two unit-norm vectors are calculated as $a = [0.17 \ 0.69 \ 0.69 \ 0.17]^T$ and $b = [0.80 \ 0.48 \ 0.16 \ 0.32]^T$. The angle parameters θ_1 , θ_2 and θ_3 for the QSP circuits can be computed from vectors a and b . Fig.8 illustrates the proposed quantum circuit for computing y_3 . For comparison, Fig.9(a) shows the complete conventional QR circuit for computing y_3 , while Fig.9(b) details its state evolution component. After executing both circuits 1000 times, the resulting measurement histograms are displayed in Fig. 10. From the result in Fig.10(a), we have the probability $p_0 = 0.403$, so the signal y_3 is $c_h c_x \sqrt{p_0} = 2.312$. From the result in Fig.10(b), we have the probability $z = 0.404$, so the signal y_3 is $c_h c_x \sqrt{z} = 2.314$. These two signal values of y_3 are almost equal to $y_3 = h^T x = 2.3$, so quantum computation can successfully implement FIR filters. Furthermore, since the circuit in Fig. 9 requires more gates than the one in Fig. 8, the proposed QSP method offers lower complexity than the conventional QR method.

Example 2: The case of $N = 8$ ($m = 3$) is studied here. The filter coefficient vector h of mean filter is chosen as

$$h = [0.05 \ 0.1 \ 0.2 \ 0.25 \ 0.2 \ 0.15 \ 0.04 \ 0.01]^T \quad (28)$$



(a)



(b)

Fig.11 The experimental results of the proposed QSP circuit method in the case of $N=8$ ($m=3$). (a) The quantum circuit. (b) The histogram of the measured result when quantum circuit is executed 1000 times.

Let the computation of output signal y_7 be considered here, then the signal vector x is given by

$$x = [6 \ 2 \ 1 \ 4 \ 5 \ 3 \ 1 \ 2]^T \quad (29)$$

Thus, traditional digital approach gives the output signal as $y_7 = h^T x = 3.21$. Two normalization constants are then given by $c_h = 0.423$ and $c_x = 9.798$. After normalizing vectors h and x into unit-norm vectors a and b , the quantum circuit of the proposed QSP method is shown in Fig.11(a). When this circuit is executed 1000 times, the histogram of measured result is shown Fig.11(b). The zero-state probability in the histogram is $p_0 = 0.664$, so the signal y_7 is $c_h c_x \sqrt{p_0} = 3.38$. This signal value is close to the value of $y_7 = h^T x = 3.21$, so quantum computation method can successfully implement FIR filter.

V. CONCLUSIONS

In this article, a quantum state preparation circuit method has been presented to implement FIR filters. Compared with conventional methods, its primary advantage is the elimination of matrix QR decomposition and IBM Qiskit's transpilation tools, thereby significantly reducing circuit complexity. However, because this study focuses exclusively on FIR filters, exploring the quantum implementation of infinite impulse response (IIR) filters remains a promising direction for future research.

REFERENCES

- [1] A. Antoniou, *Digital Filters: Analysis, Design, and Signal Processing Applications*, McGraw-Hill, 2018.
- [2] B.M. Shankar, M. Ramkumar, and V. Saravanan, "Design of low power area efficient 2D FIR filter using optimized multiplier and adder for speech signal analysis," *Circuits, Systems, and Signal Processing*, Vol. 44, pp. 1138-1163, 2025.
- [3] H.Y. Wong, *Quantum Computing Architecture and Hardware for Engineers*, Springer, 2025.
- [4] S. Y.C. Chen, "Quantum artificial intelligence: from quantum neural networks to self-programming architectures," *IEEE Circuits and Systems Magazine*, First Quarter, pp. 41-66, 2026.
- [5] C.C. Tseng and S.L. Lee, "Finite impulse response digital filter implementation using quantum computation and orthogonal triangular decomposition," *Engineering Proceedings*, Vol. 134, pp. 1-12, 2026.
- [6] S.L. Lee and C.C. Tseng, "A closed-form parameter determination method for quantum state preparation circuit," in *Proc. IEEE ISAPCS 2024*, pp. 1-5, Kaohsiung, Taiwan, December 2024.