Flag fault-tolerant error correction, measurement, and quantum computation for cyclic Calderbank-Shor-Steane codes

Theerapat Tansuwannont [1,*] Christopher Chamberland,[2,1,†] and Debbie Leung [3,4,‡]

[1]*Institute for Quantum Computing and Department of Physics and Astronomy, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1*

[2]*IBM T. J. Watson Research Center, Yorktown Heights, New York 10598, USA*

[3]*Institute for Quantum Computing and Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1*

[4]*Perimeter Institute for Theoretical Physics, Waterloo, Ontario, Canada N2L 2Y5*

(Received 26 July 2019; published 24 January 2020)

Flag qubits have recently been proposed in syndrome extraction circuits to detect high-weight errors arising from fewer faults. The use of flag qubits allows the construction of fault-tolerant protocols with the fewest number of ancillas known to date. In this work, we prove some critical properties of Calderbank-Shor-Steane (CSS) codes constructed from classical cyclic codes that enable the construction of a flag fault-tolerant error correction scheme. We then develop fault-tolerant protocols as well as a family of circuits for flag fault-tolerant error correction and operator measurement, requiring only four ancilla qubits and applicable to cyclic CSS codes of distance 3. The measurement protocol can be further used for logical Clifford gate implementation via quantum gate teleportation. We also provide examples of cyclic CSS codes with large encoding rates.

DOI: 10.1103/PhysRevA.101.012342

I. INTRODUCTION

Fault-tolerant quantum computation is an essential component in building a large scale quantum computer. It enables *arbitrarily* low logical error rates, despite all operations (including those used to perform error correction) may be noisy, as long as the noise strength is below a *constant but sufficiently small* threshold value [1–4]. The value of the threshold depends on several factors, including the underlying quantum error correcting code, the design of the fault-tolerant gadgets and the error correction protocol, the speed of quantum measurements and classical processing of the error syndromes, and the underlying physical noise. Currently, the surface code appears to be a strong candidate for fault-tolerant quantum computation given its high threshold value as well as the geometric locality of the gates used in the syndrome extraction circuits [5–9].

Meanwhile, low logical error rates requires large qubit and gate overheads [10–12]. Therefore, a fault-tolerant protocol that uses fewer ancilla qubits (and thus lower overheads) is easier to realize experimentally. A fault-tolerant protocol limits the number of physical errors in each code block arising from a single fault. Recently, Chao and Reichardt [13,14] showed that fault-tolerant error correction (FTEC) as well as fault-tolerant quantum computation can be achieved using only two extra ancilla qubits for perfect distance-3 codes. The idea is to use *flag qubits* to detect high-weight errors arising from a single fault. Furthermore, Reichardt showed that stabilizer measurements with flag qubits for the Steane code can be parallelized to reduce the circuit depths [15]. In Ref. [16] FTEC protocol using very few flag qubits were developed for several families of stabilizer codes of arbitrary distance. For example, color codes with a hexagonal lattice and arbitrary distance require only four ancilla qubits in the FTEC scheme. The protocol in Ref. [16] can be used with LDPC (low density parity check) codes to achieve constant overhead [17–19]. Flag qubits were further used for fault-tolerant preparation of magic states with very low overhead compared to previous distillation schemes when Clifford gates are noisy [20]. Last, in Ref. [21], it was shown how flag qubits can be used to fault-tolerantly prepare GKP states.

The idea behind flag-FTEC [13] is that high-weight errors arising from a single fault have special structure. Despite their high weight, these errors can be alerted using few flag qubits and distinguished by subsequent syndrome measurements. However, there is no general theory what codes admit the flag technique. An interesting family of quantum codes consists of Calderbank-Shor-Steane (CSS) codes constructed from classical cyclic codes. These codes have cyclic structures, each stabilizer generator is either X type or Z type, and some of these codes have high encoding rates. These properties make them a good choice for fault-tolerant quantum computation (see Sec. VII).

In this work, we generalize the flag technique to the family of cyclic CSS codes by exploiting the cyclic structure in the high-weight errors arising from a single fault. We build on the previous flag-FTEC schemes and obtain a flag-FTEC scheme applicable to cyclic CSS codes of distance 3. In particular, we construct circuits for measuring the error syndromes using flag qubits which require only four ancilla qubits (see Fig. 3). The circuit uses a particular ordering of the controlled-NOT (CNOT) gates which is independent of the underlying stabilizer code. Our work further expands the code

*ttansuwannont@uwaterloo.ca

†christopher.chamberland@ibm.com

‡wcleung@uwaterloo.ca

families where flag-FTEC schemes can be used with very few ancilla qubits. Moreover, the number of required ancilla qubits is independent of the weights of the stabilizers being measured. Finally, we provide a flag fault-tolerant (flag-FT) operator measurement protocol for cyclic CSS codes, which can be further used for Clifford gate implementation and other applications.

The paper is organized as follows: In Sec. II we review the basic properties of flag error correction and CSS codes. Key definitions which are used in several parts of the paper are introduced. We define the notion of distinguishable errors and consecutive error sets which are key components of our flag-FTEC scheme. We conclude the section by stating the consecutive error lemma for general CSS codes (Lemma 1), an important building block for constructing our flag-FTEC scheme. In Sec. III we review basic properties of classical cyclic codes, then state the cyclic permutation lemma (Lemma 2) and the consecutive error lemma for cyclic CSS codes (Lemma 3). Using the lemmas, we state and prove the error-distinguishability theorem (Theorem 2) which is the final ingredient required to construct our flag-FTEC scheme for CSS codes constructed from classical cyclic codes. In Sec. IV we describe the syndrome extraction circuit used in our flag-FTEC protocol and proceed by describing the protocol in detail as well as explaining how it satisfies the fault-tolerance criteria. In Sec. V we provide a flag-FT measurement protocol for Pauli operators, and its possible applications are discussed in Sec. VI. Examples of distance-3 cyclic CSS codes are given in Sec. VII. Last, we discuss our results and directions for future work in Sec. VIII.

II. FLAG ERROR CORRECTION WITH CSS CODES

CSS codes form one of the most studied families of quantum codes since they have nice properties for fault-tolerant quantum computation. It has been shown recently that the technique of flag-FTEC can be applied to several families of codes [13,16], but it remains open whether the techniques can also be applied to general CSS codes. In this section, we will analyze the idea behind flag techniques and provide the conditions which make CSS codes suitable for flag-FTEC in Lemma 1. This lemma will be a main ingredient for our theorem for cyclic CSS codes in the next section.

We start this section by first defining CSS codes (readers who are familiar with quantum error correcting codes in the stabilizer formalism may skip the following paragraphs to the end of Theorem 1). CSS codes are constructed from classical binary linear codes [22] as follows: An $[n, k, d]$ classical linear code C encodes k bits in n and has distance d (the minimum Hamming weight of the codewords). It corrects up to $t = \lfloor (d-1)/2 \rfloor$ errors. The code is defined by the parity check matrix H which consists of $n-k$ independent rows that are orthogonal to every codeword. The dual code $C^\perp$ of C consists of codewords that are orthogonal to all codewords in C. Note that $C^\perp$ is generated by H, that is, each codeword in $C^\perp$ is a linear combination of rows of H.

A quantum $[\![n, k, d]\!]$ stabilizer code [23,24] encodes k logical qubits in n physical qubits. It is the simultaneous $+1$ eigenspace of $n-k$ commuting, independent, Pauli operators. These Pauli operators multiplicatively generate a group called the *stabilizer group* for the code, and the Pauli operators are called the *stabilizer generators*. The code has distance d (see Ref. [23]), and it can correct errors acting on up to $t = \lfloor (d-1)/2 \rfloor$ qubits. Let I, X, Y, Z denote the single-qubit Pauli operators. A Pauli operator P on n qubits, given by $P = \bigotimes_{i=1}^{n} X^{x_i} Z^{z_i}$ up to a phase, has a *symplectic representation* $\sigma(P)$ which is the $2n$-bit string $\sigma(P) = (x_1, \ldots, x_n | z_1, \ldots, z_n)$. The symplectic representation of a stabilizer code is an $(n-k) \times 2n$ binary matrix where the ith row is the symplectic representation of the ith generator. The CSS codes first proposed in Refs. [25,26] can be defined in the stabilizer formalism as follows:

Definition 1. CSS code.

An $[\![n, k, d]\!]$ stabilizer code is a *CSS code* if the generators can be chosen such that the code has symplectic representation

$$\begin{pmatrix} A & | & 0 \\ 0 & | & B \end{pmatrix}, \tag{1}$$

where A is an $r_x \times n$ matrix and B is an $r_z \times n$ matrix for some r_x and r_z with $r_x + r_z = n - k$. A and B are called X and Z stabilizer matrices.

In other words, a CSS code is a stabilizer code whose generators can be chosen to be either tensor products of I and X or of I and Z. The generators of X type and Z type are called X and Z stabilizers, respectively. With this choice of generators, the X errors and Z errors can be detected separately.

Theorem 1. CSS code construction [24].

Let C_x be an $[n, k_x, d_x]$ classical linear code with parity check matrix H_x and C_z be an $[n, k_z, d_z]$ classical linear code with parity check matrix H_z. Suppose that $H_x^T H_z = 0$, or equivalently, $C_x^\perp \subseteq C_z$. Then the following binary matrix

$$\begin{pmatrix} H_x & | & 0 \\ 0 & | & H_z \end{pmatrix} \tag{2}$$

is the symplectic representation of an $[\![n, k, d]\!]$ stabilizer code C with $k = k_x + k_z - n$ and $d \geqslant \min\{d_x, d_z\}$.

In an EC protocol, the syndrome measurement corresponds to the measurement of all stabilizer generators. Consider an $[\![n, k, d]\!]$ CSS code which can correct errors of maximum weight $t = \lfloor (d-1)/2 \rfloor$. Each generator is either X-type or Z-type stabilizer, and it acts nontrivially on m qubits where $m \in \{1, \cdots, n\}$. We can assume that, up to qubit permutations, the stabilizer being measured is of the form $I^{\otimes n-m} \otimes X^{\otimes m}$ or $I^{\otimes n-m} \otimes Z^{\otimes m}$. The ideal circuits for measuring weight-m X stabilizers and weight-m Z stabilizers are shown in Fig. 1.

However, the EC protocol involving the aforementioned circuit has a drawback. Suppose the circuit is not perfect, and each location (a state preparation step, a gate, or a measurement) can have a fault. Suppose that $v \leqslant t$ faults happen. In some cases, these v faults can result in an error of weight greater than t in the output state of the circuit, which may not be correctable anymore. This circuit spreads errors and is not generally suitable for building EC protocols with an important property called *fault tolerance* [27], defined as follows:

Definition 2. Fault-tolerant error correction [27].

For $t = \lfloor (d-1)/2 \rfloor$, an error correction protocol using a distance-d stabilizer code C is *t-fault-tolerant* if the following two conditions are satisfied:

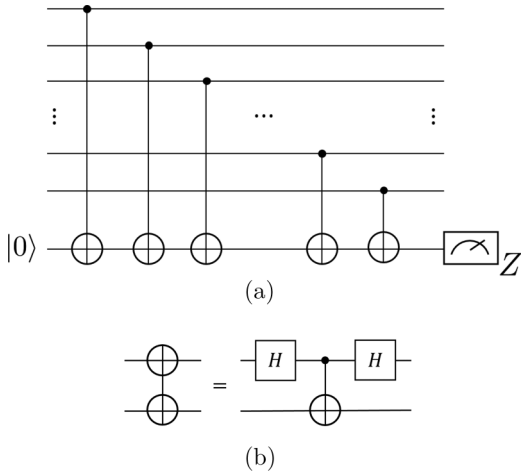

FIG. 1. (a) The ideal circuit for measuring a weight-m Z stabilizer. Only the qubits with nontrivial support on the stabilizer being measured are shown. The measurement is performed on the eigenbasis of Z operator (i.e., the computational basis), and the measurement results 0 and 1 correspond to the $+1$ and -1 eigenvalues of Z. The circuit for measuring the X stabilizers is obtain by replacing the CNOT gates with the gates is shown in panel (b).

(1) For an input codeword with error of weight v_1, if v_2 faults occur during the protocol with $v_1 + v_2 \leqslant t$, ideally decoding the output state gives the same codeword as ideally decoding the input state.

(2) For v faults during the protocol with $v \leqslant t$, no matter how many errors are present in the input state, the output state differs from a codeword by an error of at most weight v.

An error on the input state might have weight $> t$, which means that it is incorrectable. Anyhow, if the number of faults is $v \leqslant t$, the second condition in Definition 2 requires that the state after correction must differ from *any valid codeword* by an error of weight $\leqslant v$. (One possible way to construct a FTEC protocol satisfying both conditions in Definition 2 is using the minimal weight correction, defined later in Definition 9.)

Ideally decoding is equivalent to performing fault-free error correction. The conditions above are simultaneously required in order to ensure that low-weight errors do not spread and become incorrectable as well as to prevent errors from accumulating between different error correction rounds.

Generally, FTEC protocols may require many ancilla qubits to avoid the spread of errors within a code block. Chao and Reichardt introduced the idea of flag qubits in Ref. [13] to reduce the number of ancilla qubits being used in FTEC. They also provided some circuit constructions for fault-tolerant extraction of syndromes for various distance-3 perfect stabilizer codes using only two ancilla qubits. To see how the flag-FTEC works, let us examine the circuit shown in Fig. 2 which is modified from the circuit in Fig. 1(a).

A flag qubit is introduced in Fig. 2 to detect a fault that can lead to data error of weight >1. If any pair of higher-weight errors detected by the flag qubit are either equivalent (up to multiplication of some stabilizer) or have different syndromes, it is possible to construct a flag-FTEC protocol which corrects higher weight errors (that arise from a single fault) using

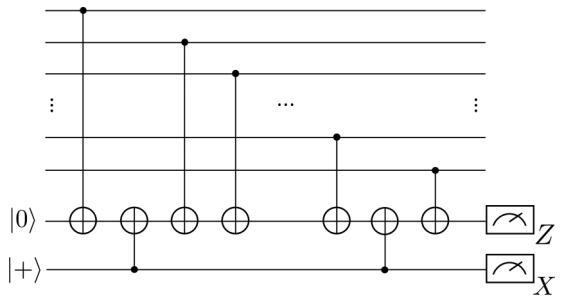

FIG. 2. A circuit obtained from Fig. 1(a) by including a flag qubit prepared in the $|+\rangle$ state. The measurement of flag qubit is performed on the eigenbasis of X operator. If a single fault produces an error of weight >1 on the data qubit, the outcome of the flag-qubit measurement will be -1, otherwise it will be $+1$.

information from the flag qubit and subsequent syndrome measurements.

The idea of flag-FTEC is further developed in Ref. [16], and the general conditions for flag-FTEC applicable to stabilizer codes of arbitrary distance are provided. In particular, the flag-FTEC condition for a stabilizer code which can correct up to one error is as follows:

Definition 3. Flag-1 FTEC condition [16].

Consider a stabilizer code generated by $\{g_1, \ldots, g_{n-k}\}$ which can correct up to one error. Let $\mathcal{E}(g_i)$ be the set of all possible errors arising from any single fault that can cause a circuit for measuring g_i to flag. For every generator g_i, all pairs of errors in $\mathcal{E}(g_i)$ must either have different syndromes or be equivalent up to multiplication of some stabilizer.

Showing that a code along with appropriate syndrome extraction circuits satisfy the general conditions for flag-FTEC can be quite challenging. Reference [16] provides a sufficient condition which implies the general flag-FTEC conditions, and a FTEC protocol using flag qubits and applicable to stabilizer codes of arbitrary distance satisfying such condition was developed. This sufficient condition can be much easier to verify, and several code families were shown to satisfy the general flag-FTEC conditions. However, not all CSS codes satisfy this sufficient condition.

As was shown in Ref. [13], for codes which do not satisfy the sufficient condition in Ref. [16], errors are spread in the measurement circuits in a way that depends on which stabilizer generators are measured, and also on the ordering of the CNOT gates used in the measurement circuits for these generators. Therefore, these specific designs in the protocol may affect the fulfillment of the flag-FTEC conditions. With an appropriate permutation of the CNOT gates of the syndrome extraction circuits, Chao and Reichardt proved that the family of $[\![2^r - 1, 2^r - 1 - 2r, 3]\!]$ quantum Hamming codes satisfied the flag-1 FTEC condition. In this work, we prove some properties of cyclic CSS codes and show that it is possible to construct syndrome extraction circuits which satisfy the flag-1 FTEC condition in Definition 3 for cyclic CSS codes of distance 3.

Reference [16] develops the notation of t-flag circuits and shows that any flag-FTEC protocol will require the use of them. We generalize their definition as follows:

Definition 4. t-flag circuit.

Let C be an $[\![n, k, d]\!]$ stabilizer code with generators $g_1, g_2 \ldots, g_{n-k}$, P be a weight-m Pauli operator which commutes with all g_i, and $\mathcal{C}(P)$ be a circuit that implements a projective measurement of P in the absence of faults. We say that $\mathcal{C}(P)$ is a *t-flag circuit* if all of the following holds:

(1) The circuit does not flag without faults and

(2) The circuit flags whenever a set of $v \leqslant t$ faults in $\mathcal{C}(P)$ leads to an error E on the output with $\min_Q[\mathrm{wt}(EQ)] > v$ where the minimization is over $Q \in \langle P, g_1, \ldots, g_{n-k}\rangle$, the group generated multiplicatively by P and the stabilizer generators g_i.

In this paper, we will use certain properties of cyclic CSS codes to develop a flag-FTEC protocol. In particular, a *single* fault in the syndrome extraction circuits of CSS codes produces errors with special properties which allow us to distinguish *consecutive* errors. To proceed with the analysis, we introduce some useful definitions and lemmas. We start by the definition of distinguishable errors as follows:

Definition 5. Distinguishable errors.

Let C be an $[\![n, k, d]\!]$ stabilizer code and let E_1 and E_2 be n-qubit Pauli errors with syndromes $s(E_1)$ and $s(E_2)$. We say that E_1 and E_2 are *distinguishable* by C if $s(E_1) \neq s(E_2)$. Otherwise we say that they are *indistinguishable*. In addition, if any pair of errors from an error set $\mathcal{E}$ are distinguishable by C, we says that $\mathcal{E}$ is distinguishable by C.

The circuit in Fig. 2 is a one-flag circuit since it will flag (the flag-qubit measurement outcome is -1) if there is a single fault causing data error of weight >1. From the flag-FTEC condition in Definition 3, our goal is to distinguish all possible higher-weight errors by subsequent stabilizer measurements. Note that the set of higher-weight errors depends on the choice of generators and the permutation of CNOT gates, and only some choices and permutations will lead to a distinguishable error set. Some CSS codes that satisfy the sufficient condition in Ref. [16] can be used in a flag-FTEC protocol.[1] However, whether flag-FTEC techniques can be applied to general CSS codes is still unknown.

Observe that permuting the CNOT gates in the measurement circuit is equivalent to permuting columns of the stabilizer matrices. In order to find CSS code families such that flag-FTEC techniques can be used, we will consider fixing the CNOT gates of syndrome extraction circuits in the *normal permutation* (i.e., applying CNOT gates from top to bottom as in Fig. 2).[2] Subsequently, we will find conditions that need to be satisfied by the X and Z stabilizer matrices.

Assume that a faulty CNOT gate can cause a two-qubit error of the form $P_1 \otimes P_2$ where $P_1, P_2 \in \{I, X, Y, Z\}$ are Pauli errors on the control and the target qubits, respectively. Consider a circuit for measuring stabilizers of the form $I^{\otimes n-m} \otimes Z^{\otimes m}$ with the normal permutation of CNOT gates as in Fig. 2 where $m \in \{1, \ldots, n\}$. A single fault at a CNOT location can result in the following types of errors:

(a) If an error from a faulty CNOT gate is of the form $P_1 \otimes P_2$ where $P_1 \in \{I, X, Y, Z\}$ and $P_2 \in \{I, X\}$, then the data error is of weight $\leqslant 1$ and the flag outcome is $+1$.

(b) If an error from a faulty CNOT gate is $P_1 \otimes P_2$ where $P_1 = I$ and $P_2 \in \{Y, Z\}$, the data error is of the form $I^{\otimes n-m+c} \otimes Z^{\otimes m-c}$ where $c \in \{1, \ldots, m\}$. In the cases where the data error has weight >1, the flag outcome is -1.

(c) If an error from a faulty CNOT is $P_1 \otimes P_2$ where $P_1 \in \{X, Y, Z\}$ and $P_2 \in \{Y, Z\}$, the data error is of the form $I^{\otimes n-m+c-1} \otimes P_1 \otimes Z^{\otimes m-c}$ where $c \in \{1, \ldots, m\}$. In the cases where the data error has weight >1, the flag outcome is -1.

Data errors of the form (b) or (c) arise due to the propagation of Z errors from the target to control qubit of CNOT gates. In addition, if a faulty CNOT gate causes the error $Z \otimes Z$, this can be viewed as an error $I \otimes Z$ caused by the preceding CNOT gate. Let $\mathcal{E}_+$ and $\mathcal{E}_-$ be sets of errors corresponding to the flag outcome $+1$ and -1, respectively. Consider an $[\![n, k, d]\!]$ CSS code C constructed from two classical codes C_x and C_z as in Theorem 1. It is obvious that $\mathcal{E}_+$ is distinguishable by C if $d \geqslant 3$. The distinguishability of errors of the form (b) in $\mathcal{E}_-$ depend on the classical code C_x. Also, any error of the form (c) in $\mathcal{E}_-$ can be considered as a product of an error of the form (b) and a weight-1 X-type error. Therefore, if the distance of C_z is $d_z \geqslant 3$ and the code C_x can distinguish all errors in the the form (b), then $\mathcal{E}_-$ is distinguishable by C. The same argument can also be applied to circuits for measuring X stabilizers.

We can see that the ability of the code to distinguish errors of the form (b) is crucial in a flag-FTEC protocol. In order to develop a flag-FTEC protocol for cyclic CSS codes, the following definitions will be very useful:

Definition 6. Left cyclic shift.

Let $P = P_1 \otimes \cdots \otimes P_n$ be an n-qubit Pauli operator and $l \in \{0, 1, \ldots, n-1\}$. The *l-qubit left cyclic shift* of the operator P, denoted by $\mathcal{L}(P, l)$, is defined as

$$\mathcal{L}(P, 0) = P, \tag{3}$$

$$\mathcal{L}(P, l) = P_{l+1} \otimes \cdots \otimes P_n \otimes P_1 \otimes \cdots \otimes P_l \quad \text{for} \quad l \neq 0. \tag{4}$$

Definition 7. Consecutive error set.

Let n be the number of qubits and $l \in \{0, 1, \ldots, n-1\}$. A *consecutive-X error set* $\mathcal{E}^x_{l,n}$ and a *consecutive-Z error set* $\mathcal{E}^z_{l,n}$ are sets of the form

$$\mathcal{E}^x_{l,n} = \{\mathcal{L}(I^{\otimes n-p} \otimes X^{\otimes p}, l) : \ p \in \{0, 1, \ldots, n-1\}\}, \tag{5}$$

$$\mathcal{E}^z_{l,n} = \{\mathcal{L}(I^{\otimes n-p} \otimes Z^{\otimes p}, l) : \ p \in \{0, 1, \ldots, n-1\}\}. \tag{6}$$

A *consecutive error product set* $\mathcal{E}^P_{l,n}$ is defined as

$$\mathcal{E}^P_{l,n} = \left\{E_x \cdot E_z : E_x \in \mathcal{E}^x_{l,n}, E_z \in \mathcal{E}^z_{l,n}\right\}. \tag{7}$$

In order to distinguish all errors in each consecutive error set, the X and Z stabilizer matrices must satisfy the conditions in the following lemma:

[1]Note that for such codes, the order of the CNOT gates in a t-flag circuit is not important.

[2]Note that for some specific codes, it is certainly possible to find circuits with fewer ancilla qubits by choosing an appropriate permutation of the CNOT gates.

Lemma 1. Consecutive error lemma (general CSS version). Let C be a CSS code constructed from the classical cyclic codes C_x and C_z following Theorem 1 with parity check matrices H_x and H_z of the form

$$H_x = \begin{pmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,n} \\ x_{2,1} & x_{2,2} & \dots & x_{2,n} \\ \dots & & & \dots \\ x_{r_x,1} & x_{r_x,2} & \dots & x_{r_x,n} \end{pmatrix}, \tag{8}$$

$$H_z = \begin{pmatrix} z_{1,1} & z_{1,2} & \dots & z_{1,n} \\ z_{2,1} & z_{2,2} & \dots & z_{2,n} \\ \dots & & & \dots \\ z_{r_z,1} & z_{r_z,2} & \dots & z_{r_z,n} \end{pmatrix}, \tag{9}$$

and let $\mathcal{E}^x_{0,n}$, $\mathcal{E}^z_{0,n}$, and $\mathcal{E}^P_{0,n}$ be consecutive-X error set, consecutive-Z error set, and consecutive error product set, respectively. Then,

(1) $\mathcal{E}^z_{0,n}$ is distinguishable by C iff for all $p, q \in \{0, \dots, n-1\}$ such that $p > q$, there exists $i \in \{1, \dots, r_x\}$ such that $x_{i,n-p+1} \oplus \dots \oplus x_{i,n-q} = 1$.

(2) $\mathcal{E}^x_{0,n}$ is distinguishable by C iff for all $p, q \in \{0, \dots, n-1\}$ such that $p > q$, there exists $i \in \{1, \dots, r_z\}$ such that $z_{i,n-p+1} \oplus \dots \oplus z_{i,n-q} = 1$.

(3) $\mathcal{E}^P_{0,n}$ is distinguishable by C iff both $\mathcal{E}^z_{0,n}$ and $\mathcal{E}^x_{0,n}$ are distinguishable by C.

Proof idea. Consider a consecutive-Z error of the form $E_p = I^{\otimes n-p} \otimes Z^{\otimes p}$. The outcome from the measurement of the ith X-type generator can be written as a sum of the last p elements of the ith row of H_x. Since two errors E_p, E_q are distinguishable iff their syndromes are not equal, the condition in statement 1 must hold. The proof of statement 2 is similar to that of statement 1, except that the consecutive errors are of X-type. Statement 3 comes from the fact that CSS codes can detect X-type and Z-type errors separately. A full proof of Lemma 1 is given in Appendix.

Note that consecutive error sets in Lemma 1 are defined on n qubits. In particular, for any subset of m out of n qubits, the consecutive error sets defined on this subset are distinguishable iff the submatrices of H_x and H_z corresponding to measurements on these m qubits satisfy similar conditions. In the next section, we will show that the cyclic symmetry of cyclic CSS codes can simplify the conditions in Lemma 1.

III. CYCLIC CSS CODES AND ERROR DISTINGUISHABILITY

In Sec. II the conditions for distinguishing errors in the consecutive error sets are given in the consecutive error lemma for general CSS codes (Lemma 1). Notice that there are some sufficient conditions for distinguishability in statements 1 and 2 that are similar, different only by some qubit shift. It is possible to simplify Lemma 1 if the CSS code has cyclic symmetry. In this section we begin by stating the definition of classical cyclic codes and outlining some of their properties. Afterwards, the cyclic permutation lemma (Lemma 2) and the consecutive error lemma for cyclic CSS codes (Lemma 3) will be provided, and the error-distinguishability theorem (Theorem 2) which is the main theorem in this work will be proved.

Definition 8. Classical cyclic code [22].

Let C be a classical binary linear code of length n. C is *cyclic* if any cyclic shift of a codeword is also a codeword, i.e., if $(c_1, c_2, \dots, c_n)$ is in a codeword, then so is $(c_n, c_1, \dots, c_{n-1})$.

Let C be a classical cyclic code of length n. There exists a unique generator polynomial $g(x) = \sum_{i=0}^{\alpha} g_i x^i$ which is also a unique monic polynomial of minimal degree in C such that C is generated by the generator matrix

$$\begin{pmatrix} g_0 & g_1 & g_2 & \dots & g_\alpha & 0 & \dots & 0 \\ 0 & g_0 & g_1 & \dots & g_{\alpha-1} & g_\alpha & \dots & 0 \\ \dots & & & & & & & \dots \\ 0 & \dots & g_0 & \dots & & & \dots & g_\alpha \end{pmatrix}. \tag{10}$$

The polynomial $h(x) = (x^n - 1)/g(x) = \sum_{i=0}^{\beta} h_i x^i$ is called the check polynomial of C. The parity check matrix of C is

$$\begin{pmatrix} h_\beta & h_{\beta-1} & \dots & h_1 & h_0 & 0 & \dots & 0 \\ 0 & h_\beta & \dots & h_2 & h_1 & h_0 & \dots & 0 \\ \dots & & & & & & & \dots \\ 0 & \dots & h_\beta & \dots & & & \dots & h_0 \end{pmatrix}. \tag{11}$$

It is known that any classical Hamming code can be made cyclic [22]. Thus, a cyclic CSS code can be constructed from permuting columns of a quantum Hamming code's stabilizer matrices. In Ref. [28] it was shown how to construct a cyclic CSS code from two classical cyclic codes.

By the symmetries of a cyclic code, we can show in the following lemma that the left cyclic shift of operators in the generating set also generates the same code.

Lemma 2. Cyclic permutation lemma.

Let C be a CSS code constructed from the classical cyclic codes C_x and C_z following Theorem 1. Suppose that the stabilizer group of C can be generated by $\{g_1, g_2, \dots, g_{n-k}\}$, then the stabilizer group of C can also be generated by $\{\mathcal{L}(g_1, l), \mathcal{L}(g_2, l), \dots, \mathcal{L}(g_{n-k}, l)\}$ for any $l \in \{0, 1, \dots, n-1\}$.

Proof idea. X-type (or Z-type) generators correspond to the parity check matrix of C_x (or C_z) which generates $C_x^\perp$ (or $C_z^\perp$). Since the dual code of a cyclic code is also cyclic, a set of cyclic permutations of generators also generates the same code. A full proof of Lemma 2 is given in Appendix.

In the previous section, the consecutive error lemma for general CSS codes (Lemma 1) gives sufficient and necessary conditions for a CSS code to be able to distinguish all errors in the consecutive error sets. The conditions can be simplified by using the symmetry of cyclic codes as follows:

Lemma 3. Consecutive error lemma (cyclic CSS version).

Let C be a CSS code constructed from the classical cyclic codes C_x and C_z (following Theorem 1) with parity check matrices H_x and H_z,

$$H_x = \begin{pmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,n} \\ x_{2,1} & x_{2,2} & \dots & x_{2,n} \\ \dots & & & \dots \\ x_{r_x,1} & x_{r_x,2} & \dots & x_{r_x,n} \end{pmatrix}, \tag{12}$$

$$H_z = \begin{pmatrix} z_{1,1} & z_{1,2} & \dots & z_{1,n} \\ z_{2,1} & z_{2,2} & \dots & z_{2,n} \\ \dots & & & \dots \\ z_{r_z,1} & z_{r_z,2} & \dots & z_{r_z,n} \end{pmatrix}. \tag{13}$$

Let $l \in \{0, 1, \ldots, n-1\}$, and let $\mathcal{E}^x_{l,n}$, $\mathcal{E}^z_{l,n}$, and $\mathcal{E}^P_{l,n}$ be consecutive-X error set, consecutive-Z error set, and consecutive error product set, respectively. Then,

(1) $\mathcal{E}^z_{l,n}$ is distinguishable by C iff for all $u_x \in \{2, \ldots, n\}$, there exists $i \in \{1, \ldots, r_x\}$ such that $x_{i,u_x} \oplus \cdots \oplus x_{i,n} = 1$.

(2) $\mathcal{E}^x_{l,n}$ is distinguishable by C iff for all $u_z \in \{2, \ldots, n\}$, there exists $i \in \{1, \ldots, r_z\}$ such that $z_{i,u_z} \oplus \cdots \oplus z_{i,n} = 1$.

(3) $\mathcal{E}^P_{l,n}$ is distinguishable by C iff both $\mathcal{E}^z_{l,n}$ and $\mathcal{E}^x_{l,n}$ are distinguishable by C.

Proof idea. The cyclic symmetry of C_x and C_z can simplify Lemma 1, resulting in fewer sufficient conditions for distinguishability in statements 1 and 2; we can fix q in Lemma 1 to be 0 and choose $u = n - p + 1$. In other words, the cyclic symmetry reduces the number of error pairs in consecutive error sets to be distinguished. This proves the statements for $l = 0$. Moreover, using Lemma 2, we can extend all statements to consecutive error sets of any $l \in \{0, 1, \ldots, n-1\}$. A full proof of Lemma 3 is given in Appendix.

Now we are ready to prove a main theorem in this work.

Theorem 2. Error-distinguishability theorem.

Let C be an $[\![n, k, d]\!]$ CSS code constructed from the $[n, k_x, d_x]$ classical cyclic code C_x and the $[n, k_z, d_z]$ classical cyclic code C_z, $l \in \{0, 1, \ldots, n-1\}$, and $\mathcal{E}^P_{l,n}$ be a consecutive error product set. If both $d_x, d_z \geqslant 3$, then $\mathcal{E}^P_{l,n}$ is distinguishable by C.

Proof. Suppose by contradiction that $\mathcal{E}^P_{l,n}$ is not distinguishable by C. Then at least one of $\mathcal{E}^z_{l,n}$ and $\mathcal{E}^x_{l,n}$ is not distinguishable by C. Similar analysis applies to either case, so, suppose $\mathcal{E}^z_{l,n}$ is not distinguishable by C. We next invoke the consecutive error lemma for cyclic CSS codes (Lemma 3), and to do so, let the cyclic code C_x has parity check matrix

$$H_x = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \cdots & & & \cdots \\ x_{r_x,1} & x_{r_x,2} & \cdots & x_{r_x,n} \end{pmatrix}, \tag{14}$$

where $x_{i_1+1,(j+1) \pmod n} = x_{i_1,j}$ for all $i_1 \in \{1, \ldots, r_x - 1\}$ and $j \in \{1, \ldots, n\}$.

By statement 1 of Lemma 3, we know that $\mathcal{E}^z_{l,n}$ is indistinguishable by C iff there exists $u_x \in \{2, 3, ..., n\}$ such that $x_{i,u_x} \oplus \cdots \oplus x_{i,n} = 0$ for all $i \in \{1, \cdots, r_x\}$; i.e., there exists a pair of errors in $\mathcal{E}^z_{l,n}$ which cannot be distinguished by any generator of C.

For $i = 1$, we have

$$x_{1,u_x} \oplus \cdots \oplus x_{1,n} = 0\,. \tag{15}$$

Using the cyclic permutation lemma (Lemma 2), we obtain a generating set for C where each generator is the one-qubit left cyclic shift of the old one. Applying the above to this generator set, we have

$$x_{1,u_x+1} \oplus \cdots \oplus x_{1,n} \oplus x_{1,1} = 0. \tag{16}$$

Now repeating the left cyclic shifts gives

$$\begin{aligned} x_{1,u_x+2} \oplus \cdots \oplus x_{1,n} \oplus x_{1,1} \oplus x_{1,2} &= 0, \\ &\vdots \\ x_{1,u_x-1} \oplus x_{1,1} \oplus \cdots \oplus x_{1,n-1} &= 0\,. \end{aligned} \tag{17}$$

From Eqs. (15) and (16), $x_{1,1} = x_{1,u_x}$; from Eqs. (16) and (17), $x_{1,2} = x_{1,u_x+1}$, and so on, until we obtain $x_{1,n} = x_{1,u_x+(n-1) \pmod n}$ (in other words, $x_{1,j} = x_{1,(u_x-1+j) \pmod n}$ for all $j \in \{1, \ldots, n\}$).

Let $w_x = \mathrm{GCD}(u_x - 1, n)$, the greatest common divisor of $u_x - 1$ and n. The conditions become

$$x_{1,j} = x_{1,j+w_x} = x_{1,j+2w_x} = \cdots = x_{1,j+n-w_x}, \tag{18}$$

for all $j \in \{1, \ldots, w_x\}$. Repeating the above steps for all i, we obtain

$$x_{i,j} = x_{i,j+w_x} = x_{i,j+2w_x} = \cdots = x_{i,j+n-w_x}, \tag{19}$$

for all $i \in \{1, \ldots, r_x\}$, $j \in \{1, \ldots, w_x\}$.

From the above, we see that any error of the form $Z_{l_x} Z_{l_x+w_x}$ (where $l_x \in \{1, \ldots, n - w_x\}$) commutes with all stabilizer generators. Now let us consider two cases:

Case 1: At least one operator of the form $Z_{l_x} Z_{l_x+w_x}$ is not in the stabilizer. In this case, the distance d of the code C is at most two. Since $d \geqslant \min\{d_x, d_z\}$ (see the CSS construction in Theorem 1), this contradicts our assumption that both $d_x, d_z \geqslant 3$.

Case 2: All operators of the form $Z_{l_x} Z_{l_x+w_x}$ are in the stabilizer. In this case, there exists a set of coefficients $a_1, \ldots, a_{r_z} \in \{0, 1\}$ such that $(g^z_1)^{a_1} \cdots (g^z_{r_z})^{a_{r_z}} = Z_{l_x} Z_{l_x+w_x}$, where g^z_i is the Z-type generator corresponding to the ith row of H_z. This means that the Z part of $\sigma(Z_{l_x} Z_{l_x+w_x})$ is a codeword in $C_z^\perp$. Since $C_z^\perp \subseteq C_x$ by the construction of CSS codes, we have that the Z part of $\sigma(Z_{l_x} Z_{l_x+w_x})$ is a codeword in C_x. Because the distance of classical codes is given by the minimum Hamming weight of the codewords, we have that $d_x \leqslant 2$ which contradicts our assumption that $d_x \geqslant 3$. ■

Although consecutive error product set $\mathcal{E}^P_{l,n}$ is distinguishable by any cyclic CSS code satisfying Theorem 2, we cannot construct an FTEC protocol using the circuit in Fig. 2 directly since the possible errors might not be in the consecutive form without qubit permutation. Moreover, permuting qubits will break the cyclic symmetry and $\mathcal{E}^P_{l,n}$ might no longer be distinguishable. In the next section, we will use Theorem 2 to find a one-flag circuit for distance-3 cyclic CSS codes that can be used in a fault-tolerant protocol satisfying both FTEC conditions in Definition 2. We point out that since p, q in the consecutive error lemma for general CSS codes (Lemma 1) are chosen to be in the set $\{0, \ldots, n-1\}$, if a cyclic CSS code can correct errors of weight $\leqslant t$, then the flag circuits should be designed such that if there are $\leqslant t$ faults during the FTEC protocol, an error of weight n cannot occur.

IV. FAULT-TOLERANT ERROR CORRECTION PROTOCOL FOR DISTANCE-3 CYCLIC CSS CODES

Fault-tolerant error correction is one of the most important building blocks for fault-tolerant quantum computation. In this section, a flag-FTEC protocol for distance-3 cyclic CSS codes is developed.[3] A one-flag circuit for cyclic CSS codes of distance 3 which is required for the flag-FTEC protocol

[3]Note that our protocol and circuit can also be applied to higher distance codes if we only consider correcting errors introduced by at most one fault.

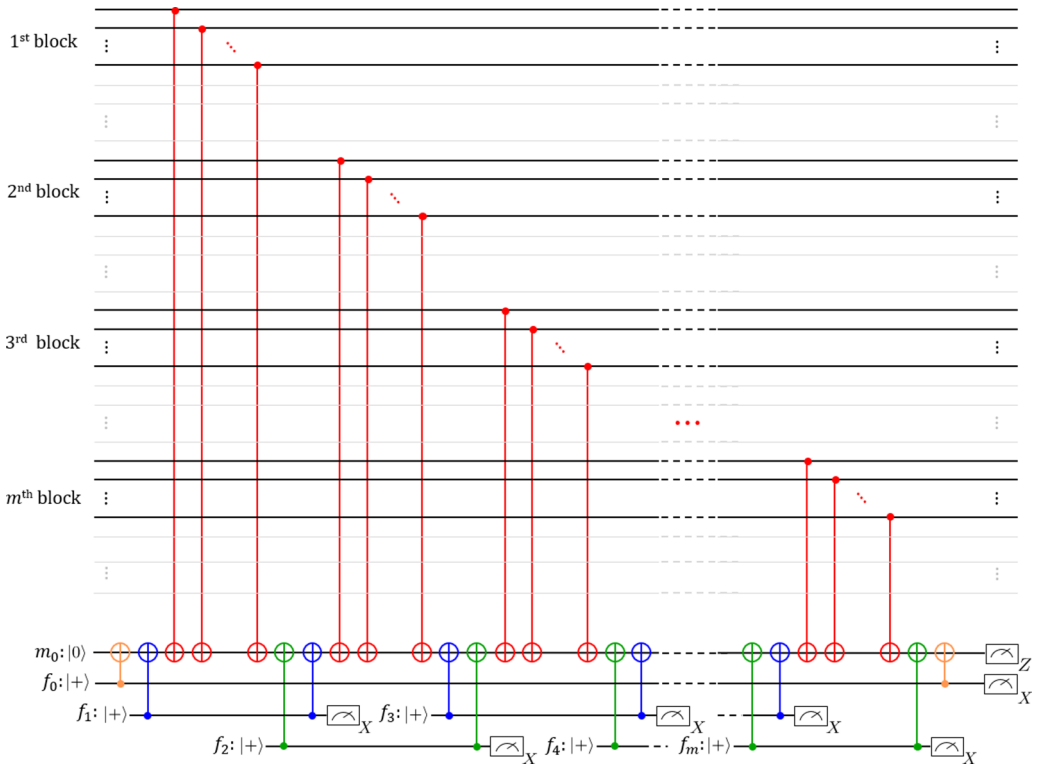

FIG. 3. Illustration of a one-flag circuit applicable to distance-3 cyclic CSS codes. The circuit measures stabilizers of the form $Z^{\otimes a_1} \otimes I^{\otimes b_1} \otimes Z^{\otimes a_2} \otimes I^{\otimes b_2} \otimes \cdots \otimes Z^{\otimes a_m} \otimes I^{\otimes b_m}$. The flag qubits are represented by the labels $f_1, \ldots, f_m$. Information from the flag outcomes along with the protocol given in Sec. IV enable the construction of a flag-FTEC protocol which satisfies both FTEC conditions in Definition 2. (For grayscale version, *red* CNOT gates are CNOT gates connecting between a data qubit and qubit m_0. The *orange*, *blue*, and *green* CNOT gates have control qubits f_0, f_i for odd i, and f_i for even i, respectively).

is provided in Fig. 3 (see Definition 4 for the definition of a t-flag circuit). Here we adapt the idea of localizing circuit faults from Ref. [13].

Suppose that the stabilizer generator being measured is of the form

$$P = Z^{\otimes a_1} \otimes I^{\otimes b_1} \otimes Z^{\otimes a_2} \otimes I^{\otimes b_2} \otimes \cdots \otimes Z^{\otimes a_m} \otimes I^{\otimes b_m},$$

where $a_i > 0$ and $b_i \geqslant 0$ are integers. The ith subblock consists of a_i qubits, which are from the $\sum_{j=1}^{i-1}(a_j + b_j) + 1$th qubit to the $\sum_{j=1}^{i-1}(a_j + b_j) + a_i$th qubit.

Notice that the blue, green and orange CNOT gates in the circuit of Fig. 3 always come in pairs. This is to ensure that when fault-free, the circuit implements a projective measurement of the stabilizer without flagging. In what follows, we will refer to the first blue, green or orange CNOT of a pair as an *opening* CNOT and the second blue, green or orange CNOT as a *closing* CNOT. Given these definitions, we have the following claim:

Claim 1. Fault-flag relations.

During the measurement of $P = Z^{\otimes a_1} \otimes I^{\otimes b_1} \otimes Z^{\otimes a_2} \otimes I^{\otimes b_2} \otimes \cdots \otimes Z^{\otimes a_m} \otimes I^{\otimes b_m}$ using the circuit in Fig. 3, the following can occur:

(1) If there are no faults, none of the f_i ancilla qubits will flag.

(2) A fault at a CNOT location resulting in a ZZ error is equivalent to the prior CNOT failing resulting in an IZ error (here Z acts on the target qubit).

(3) Suppose that a fault occurs on one of the red CNOTs and causes a Z error on the ancilla m_0. If the fault occurs on subblock a_i where $i \geqslant 1$, only the ancillas f_0 and f_i will flag.

(4) Suppose that a fault occurs on a blue or green CNOT. Let the control qubit be the ancilla f_i. If it is the opening CNOT and causes a Z error on ancilla m_0, the ancillas f_0, f_i, and f_{i-1} will flag. If it is the closing CNOT and causes a Z error on the ancilla m_0, the ancillas f_0 and f_{i+1} will flag. However if the fault occurs on a blue or green CNOTs at the boundary,[4] if the opening CNOT of f_1 is faulty, f_0 and f_1 will flag, and if the closing CNOT of f_m is faulty, only f_0 will flag.

(5) A fault occurring at an orange CNOT gate will not cause a data qubit error (since a Z spreading to all qubits is equivalent to the stabilizer being measured). Furthermore, only the ancilla f_0 can flag in this case (depending on whether the error was of the form IZ or ZZ and also whether it occurred on the opening or closing orange CNOT).

From the above claim, one can verify that a single fault resulting in a data qubit error E with $\min_Q[\mathrm{wt}(EQ)] > 1$ where $Q \in \langle P, g_1, \ldots, g_{n-k} \rangle$ will always cause at least one flag qubit to flag (see the conditions to be held for a t-flag circuit in Definition 4). Thus the circuit in Fig. 3 is a one-flag circuit. Note that an analogous claim can be made for X-type stabilizers.

[4]By boundary we are referring to either the first blue CNOT after the first subblock or the last green CNOT after the mth subblock.

Before describing the FTEC protocol, we require one more definition:

Definition 9. Minimum weight correction.

Given the syndrome $s = s(E)$ of an error E, we let $E_{\min}(s)$ be a *minimal weight correction* of E.

Note that many errors can lead to the same syndrome. In particular, errors corresponding to the same syndrome differ by some multiplication of stabilizers or logical operators. If error E is correctable, applying $E_{\min}(s)$ can correct such error as we expected. However, if E is not correctable [i.e., $\mathrm{wt}(E) > t$], applying $E_{\min}(s)$ will project the codeword back to the coding subspace, but the resulting codeword may differ from the original codeword by some logical operation. This property of minimal weight correction is required so that the FTEC protocol satisfy the second FTEC condition in Definition 2.

Using the error-distinguishability theorem (Theorem 2), the fault-flag relations (Claim 1), and the definition of minimum weight correction (Definition 9), we now describe a FTEC protocol that satisfies the FTEC conditions in Definition 2 for distance-3 cyclic CSS codes using a procedure adapted from Ref. [16]. In what follows, we define $s^{(r)} = (s_x^{(r)}|s_z^{(r)})$ to be the syndrome obtained during round r (either using flag or nonflag circuits), where $s_x^{(r)}$ and $s_z^{(r)}$ are the syndromes obtained from X-type and Z-type stabilizers, respectively.

FTEC Protocol:

Let C be an $[\![n, k, d]\!]$ cyclic CSS code satisfying Theorem 2 with stabilizer group $S = \langle g_1, \ldots, g_{n-k}\rangle$. Let $\mathcal{C}(g_i)$ be the one-flag circuit of Fig. 3 for stabilizer g_i. Repeat the syndrome measurement (measurement of all stabilizer generators) using the one-flag circuits until one of the following conditions is satisfied, then perform its corresponding operations:

(1) If the syndrome is repeated twice in a row and there are no flags, apply $E_{\min}(s^{(1)})$.

(2) If there are no flags and the syndromes $s^{(1)}$ and $s^{(2)}$ differ, repeat the syndrome measurement using nonflagged circuits. Apply the correction $E_{\min}(s^{(3)})$.

(3) If f_0 does not flag but f_i flags (with $i \geqslant 1$) during round one, stop. Repeat the syndrome measurement using nonflag circuits and apply $E_{\min}(s^{(2)})$. If there are no flags in the first round but in round two f_i flags and f_0 does not flag, stop. Apply $E_{\min}(s^{(1)})$.

(4) If f_0 flags at round r anytime during the protocol, stop and do one of the following:

(a) If f_i does not flag for all $i \geqslant 1$, repeat the syndrome measurement using nonflag circuits. Apply $E_{\min}(s^{(r+1)})$.

(b) If there is only one i such that f_i flags (with $i \geqslant 1$), apply $I^{\otimes c} \otimes Z^{\otimes a_{i+1}} \otimes I^{\otimes b_{i+1}} \otimes \cdots \otimes Z^{\otimes a_m} \otimes I^{\otimes b_m}$ to the data if the stabilizer being measured is a Z stabilizer or $I^{\otimes c} \otimes X^{\otimes a_{i+1}} \otimes I^{\otimes b_{i+1}} \otimes \cdots \otimes X^{\otimes a_m} \otimes I^{\otimes b_m}$ if it is an X stabilizer, where $c = \sum_{j=1}^{i}(a_j + b_j)$. Repeat the syndrome measurement using nonflag circuits yielding syndrome $s^{(r+1)} = (s_x^{(r+1)}|s_z^{(r+1)})$.

(i) If the stabilizer being measured is a Z stabilizer and there is an element E_z in $\mathcal{E}_{l,n}^z$ where $l = n - c + b_i$ that satisfies $s(E_z) = s_x^{(r+1)}$, apply E_z followed by $E_{\min}(s_z^{(r+1)})$. Otherwise, apply $E_{\min}(s^{(r+1)})$.

(ii) If the stabilizer being measured is an X stabilizer and there is an element E_x in $\mathcal{E}_{l,n}^x$ where $l = n - c + b_i$ that satisfies $s(E_x) = s_z^{(r+1)}$, apply E_x followed by $E_{\min}(s_x^{(r+1)})$. Otherwise, apply $E_{\min}(s^{(r+1)})$.

(c) If there is an i such that f_i and f_{i+1} flag, perform the same sequence operations as in 4(b).

To see that the above protocol satisfies the FTEC conditions in Definition 2, we will assume that there is at most one fault during the protocol. If a fault in any of the CNOT gates introduces a Z error on ancilla m_0, then f_0 and at least one f_i (with $i \geqslant 1$) will flag (unless the first orange CNOT introduces an error of the form ZZ or the last orange CNOT introduces an error of the form IZ which in both cases, there will be no data qubit error). If there is only one flag during round one, either f_0 or f_i, then the fault could either have been caused by a measurement error, idle qubit error on the ancilla f_0 or f_i, or an error on the control qubit of the CNOT gate interacting with f_0 or f_i. However in all three cases, the error could not have spread to the data. By repeating the syndrome measurement and applying $E_{\min}(s^{(2)})$, both criteria of Definition 2 will be satisfied. Note that if f_i flags during round two, then the syndrome obtained during round one corresponds to the data qubit error (since there could not have been a measurement error giving the wrong syndrome during the first round), so correcting using $s^{(1)}$ will again satisfy both criteria in Definition 2.

Next, let us consider the case where none of the f_i ancillas flag. By the circuit construction, a single fault can introduce an error E with $\mathrm{wt}(E) \leqslant 1$. If the same syndrome is repeated twice in a row, then applying $E_{\min}(s^{(1)})$ can result in a data error of weight at most one. If $s^{(1)} \neq s^{(2)}$, then a fault occurred in either the first or second round. Thus repeating the syndrome measurement a third time and applying $E_{\min}(s^{(3)})$ will remove the data errors or project the code back to the coding subspace.

Next we consider the case where a fault happens on a red CNOT introducing a Z error on the ancilla m_0 and a P error on the data qubit where $P \in \{I, X, Y, Z\}$. If the fault occurs on the ith subblock, then f_0 will flag and there will be only one $i \geqslant 1$ such that f_i flags. Applying $I^{\otimes c} \otimes Z^{\otimes a_{i+1}} \otimes I^{\otimes b_{i+1}} \otimes \cdots \otimes Z^{\otimes a_m} \otimes I^{\otimes b_m}$ where $c = \sum_{j=1}^{i}(a_j + b_j)$ to the data if the stabilizer being measured is a Z stabilizer (or $I^{\otimes c} \otimes X^{\otimes a_{i+1}} \otimes I^{\otimes b_{i+1}} \otimes \cdots \otimes X^{\otimes a_m} \otimes I^{\otimes b_m}$ if it is an X stabilizer) guarantees that the resulting error is a product of Z-type error from $\mathcal{E}_{l,n}^z$ and an X-type error of weight at most 1 (or a product of X-type error from $\mathcal{E}_{l,n}^x$ and a Z-type error of weight at most 1). By the error-distinguishability theorem (Theorem 2), errors in the set $\mathcal{E}_{l,n}^z$ (or $\mathcal{E}_{l,n}^x$) can be distinguished. Thus applying the correction in 4(b) of the protocol will remove the error if there are no input errors. However, if there is an input error, then applying $E_{\min}(s^{(r+1)})$ will project the code back to the coding subspace.

Last, if a fault occurs on a blue or green CNOT, then from the fault-flag relations (Claim 1) either the case in 4(b) or 4(c) will be satisfied. However in both cases, the Z error will spread to the data in the same way. Hence the correction proposed in 4(c) will satisfy the FTEC criteria of Definition 2.

A list of possible faults during the flag-FTEC protocol and corresponding correction procedures is given in Table I.

TABLE I. Possible faults during the flag-FTEC protocol in Sec. IV and their corresponding correction procedures. We assume that the number of faults v_2 is at most 1.

Type of faults	Correction procedure
No fault	1
Qubit or measurement fault on m_0	1 or 2
Qubit or measurement fault on f_0	1 or 4(a)
Qubit or measurement fault on f_i	1 or 3
Red CNOT fault with I or X error on on the target qubit	2
Red CNOT fault with Y or Z error on the target qubit	4(b)
Blue or green CNOT fault with I or X error on the target qubit	1 or 2 or 3
Blue or green CNOT fault with Y or Z error on the target qubit	4(b) or 4(c)
Orange CNOT fault with I or X error on the target qubit	1 or 2 or 4(a)
Orange CNOT fault with Y or Z error on the target qubit	2 or 4(a)

V. FAULT-TOLERANT MEASUREMENT PROTOCOL FOR DISTANCE-3 CYCLIC CSS CODES

Besides FTEC, there are other important components for fault-tolerant computation: FT state preparation, FT measurement, and FT quantum gate implementation. In this section, we provide a flag-FT measurement protocol for distance-3 cyclic CSS codes. The measurement protocol plays an important role in fault-tolerant quantum computation on cyclic CSS codes since it can also be used as a subroutine for FT state preparation, FT quantum gate implementation, and other techniques, described later in Sec. VI.

The flag-FT protocol provided in this section is similar to the flag-FTEC protocol in Sec. IV except that the idea of consecutive error correction is developed so that it is applicable not only to stabilizer measurements but also to measurements of any Pauli operator commuting with all generators. We begin by introducing the definition of fault-tolerant nondestructive measurement adapted from Ref. [27] as follows:

Definition 10. Fault-tolerant nondestructive measurement. For $t = \lfloor (d-1)/2 \rfloor$, a nondestructive measurement protocol using a distance-d stabilizer code C is *t-fault-tolerant* if the following two conditions are satisfied:

(1) For an input codeword with error of weight v_1, if v_2 faults occur during the measurement protocol with $v_1 + v_2 \leqslant t$, ideally decoding the output state after measurement gives the same state as ideally decoding the input state and then performing ideal nondestructive measurement. The result obtained from measuring the input codeword is the same as that of an ideal measurement on the ideally decoded input state.

(2) For an input codeword with error of weight v_1, if v_2 faults occur during the measurement protocol with $v_1 + v_2 \leqslant t$, the output state differs from a codeword by an error of at most weight $v_1 + v_2$.

[Here we need to modify Definition 10 from the usual definition of fault-tolerant (destructive) measurement since we would like to obtain both measurement result and post-measurement state. These ingredients are important in the applications discussed in Sec. VI.]

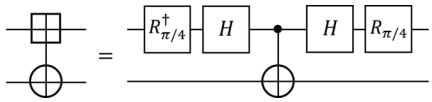

FIG. 4. Quantum gates for measuring Y operator where $R_{\pi/4} = \mathrm{diag}(1, i)$.

Suppose that the operator being measured P commutes with all generators and is of the form

$$P = P_1^{\otimes a_1} \otimes I^{\otimes b_1} \otimes P_2^{\otimes a_2} \otimes I^{\otimes b_2} \otimes \cdots \otimes P_m^{\otimes a_m} \otimes I^{\otimes b_m}, \tag{20}$$

where $a_i > 0$ and $b_i \geqslant 0$ are integers and $P_i \in \{X, Y, Z\}$. The ith subblock consists of a_i qubits acted on by $P_i^{\otimes a_i}$. A one-flag circuit for operator measurement is similar to the circuit given in Fig. 3, while the measurements of Z, X, and Y operators correspond to CNOT gates, the gates shown in Fig. 1(b), and the gates shown in Fig. 4. With the slight modification where CNOT gates are replaced by the gates for measuring $P_i \in \{X, Y, Z\}$, one can verify that the fault-flag relations (Claim 1) is also applicable in this setting.

Using the error-distinguishability theorem (Theorem 2) and the fault-flag relations (Claim 1), we now describe a flag-FT measurement protocol that satisfies the FT nondestructive measurement conditions in Definition 10 for distance-3 cyclic CSS codes. Here we define $m^{(r_1)}$ to be the measurement result obtained from operator measurement (using either flag or nonflag circuits) during round r_1, and define $s^{(r_2)} = (s_x^{(r_2)}|s_z^{(r_2)})$ to be the syndrome obtained from syndrome measurement (using either flag or nonflag circuits) during round r_2 of error correction. The protocol is as follows:

Flag-FT Operator Measurement Protocol:

Let C be an $[\![n, k, d]\!]$ cyclic CSS code satisfying Theorem 2. Let $\mathcal{C}(P)$ be the one-flag circuit of Fig. 3 for measuring a Pauli operator P of the form

$$P = P_1^{\otimes a_1} \otimes I^{\otimes b_1} \otimes P_2^{\otimes a_2} \otimes I^{\otimes b_2} \otimes \cdots \otimes P_m^{\otimes a_m} \otimes I^{\otimes b_m}, \tag{21}$$

where P commutes with all generators of C. Repeat the measurement of P using the one-flag circuits until one of the following conditions is satisfied, then perform its corresponding operations:

(1) If first two operator measurement results coincide ($m^{(1)} = m^{(2)}$) and there are no flags, perform the syndrome measurement twice using one-flag circuits for flag-FTEC.

(a) If $s^{(1)} = s^{(2)} = 0$ and there are no flags during both syndrome measurement rounds, output $m^{(1)}$.

(b) If $s^{(1)} \neq s^{(2)}$ or at least one-flag qubit flags during the syndrome measurement, apply the correction described by the flag-FTEC protocol of Sec. IV, then output $m^{(1)}$.

(c) If $s^{(1)} = s^{(2)} \neq 0$ and there are no flags during the syndrome measurement, apply the correction $E_{\min}(s^{(1)})$. Repeat the operator measurement using a nonflag circuit, then output $m^{(3)}$.

(2) If $m^{(1)} \neq m^{(2)}$ and there are no flags, perform one syndrome measurement round using nonflag circuits for error

correction and apply $E_{\min}(s^{(1)})$. Repeat the operator measurement using a nonflag circuit, then output $m^{(3)}$.

(3) If f_0 does not flag but f_i flags (with $i \geqslant 1$) during round one, stop. Repeat the operator measurement using a nonflag circuit then output $m^{(2)}$. If there are no flags during round one but f_i flags and f_0 does not flag during round two, output $m^{(1)}$.

(4) If f_0 flags at round r_1 anytime during the protocol, stop and do one of the followings:

(a) If f_i does not flag for all $i \geqslant 1$, repeat the operator measurement using a nonflag circuit and output $m^{(r_1+1)}$.

(b) If there is only one i such that f_i flags (with $i \geqslant 1$), apply $I^{\otimes c} \otimes P_{i+1}^{\otimes a_{i+1}} \otimes I^{\otimes b_{i+1}} \otimes \cdots \otimes P_m^{\otimes a_m} \otimes I^{\otimes b_m}$ to the data, where $c = \sum_{j=1}^{i} a_j + b_j$. Perform the syndrome measurement using nonflag circuits for error correction yielding syndrome $s^{(r_2)} = (s_x^{(r_2)}|s_z^{(r_2)})$.

(i) If $P_i = Z$, apply $E_z \in \mathcal{E}_{l,n}^z$ that satisfies $s(E_z) = s_x^{(r_2)}$ where $l = n - c + b_i$, followed by $E_{\min}(s_z^{(r_2)})$.

(ii) If $P_i = X$, apply $E_x \in \mathcal{E}_{l,n}^x$ that satisfies $s(E_x) = s_z^{(r_2)}$ where $l = n - c + b_i$, followed by $E_{\min}(s_x^{(r_2)})$.

(iii) If $P_i = Y$, apply $E \in \mathcal{E}_{l,n}^P$ that satisfies $s(E) = s^{(r_2)}$ where $l = n - c + b_i$.

Afterwards, repeat the operator measurement using a nonflag circuit, then output $m^{(r_1+1)}$.

(c) If there is an i such that f_i and f_{i+1} flag, perform the same sequence of operations as in 4(b).

To see that both criteria for FT nondestructive measurement in Definition 10 are satisfied, we will assume that the weight of an input error v_1 and the number of faults during the protocol v_2 satisfy $v_1 + v_2 \leqslant 1$. Similar to the FTEC protocol, f_0 and at least one f_i (with $i \geqslant 1$) will flag whenever a fault in any CNOT gate causes Z error on m_0. If there is no flags, a single fault can introduce error of weight at most one. If the measurement result is repeated twice, then there is no fault in the circuits. However, the measurement result might be incorrect due to the input error. By performing full syndrome measurement twice with flag circuits, we can determine from $s^{(1)}$ and $s^{(2)}$ whether these is no input error, there is a fault during syndrome measurement, or there is an input error of weight 1. The procedure in 1(a), 1(b), and 1(c) can correct possible errors and output the right operator measurement result with corresponding codeword after projective measurement.

Now let us consider the case that there is no flags but $m^{(1)} \neq m^{(2)}$. This is the case where a fault occurred in either the first or second round. Therefore, performing error correction and repeating the operator measurement can give the correct result.

Next, consider the case that there is only one flag, either f_0 or f_i with $i \geqslant 1$. The fault could be a measurement error, idle qubit error on the ancilla f_0 or f_i, or an error on the control qubit of the CNOT gate interacting with f_0 or f_i. Repeating the operator measurement can give the right result. Note that if f_i flags during round two, then the result obtained during round one corresponds to the right outcome.

Now let us consider the case where a fault happens on a red CNOT introducing a Z error on the ancilla m_0 and a $\tilde{P}$ error on the data qubit where $\tilde{P} \in \{I, X, Y, Z\}$. If the fault occurs on the ith subblock, then f_0 and only one f_i with $i \geqslant 1$ will flag. Applying $I^{\otimes c} \otimes P_{i+1}^{\otimes a_{i+1}} \otimes I^{\otimes b_{i+1}} \otimes \cdots \otimes P_m^{\otimes a_m} \otimes I^{\otimes b_m}$ to the data guarantees that the resulting error is in the form $I^{\otimes c-a_i-b_i} \otimes I^{\otimes c-1} \otimes \tilde{P} \otimes P_i^{\otimes a_i-c} \otimes I^{\otimes b_i} \otimes I^{\otimes n-c}$ (where $c = \sum_{j=1}^{i} a_j + b_j$). If $P_i = Z$ (or $P_i = X$), the resulting error is a product of consecutive error in $\mathcal{E}_{l,n}^z$ (or $\mathcal{E}_{l,n}^x$) and X-type error (or Z-type error) of weight one, where $l = n - c + b_i$. If $P_i = Y$, the resulting error is a consecutive error in $\mathcal{E}_{l,n}^P$. By the error-distinguishability theorem (Theorem 2), errors in $\mathcal{E}_{l,n}^x$, $\mathcal{E}_{l,n}^z$, and $\mathcal{E}_{l,n}^P$ are distinguishable. Therefore, performing a full syndrome measurement followed by appropriate error correction as in 4(b) will remove the error, and repeating the operator measurement gives the correct outcome. The case that a fault occurs on a blue or green CNOT corresponds to either 4(b) or 4(c), and the same correction procedure can be applied.

TABLE II. Possible faults during the flag-FT operator measurement protocol in Sec. V and their corresponding correction procedures. Here we assume that the number of input errors v_1 and the number of faults v_2 satisfy $v_1 + v_2 \leqslant 1$.

Type of faults	Correction procedure
No fault during operator measurement	
No input error, no fault during syndrome measurement	1(a)
No input error, one fault during syndrome measurement	1(b) or 1(c)
Weight-1 input error, no fault syndrome measurement	1(c)
One fault during operator measurement	
Qubit or measurement fault on m_0	2
Qubit or measurement fault on f_0	4(a)
Qubit or measurement fault on f_i	3
Red CNOT fault with I or X error on the target qubit	2
Red CNOT fault with Y or Z error on the target qubit	4(b)
Blue or green CNOT fault with I or X error on the target qubit	1(a), 2, or 3
Blue or green CNOT fault with Y or Z error on the target qubit	4(b) or 4(c)
Orange CNOT fault with I or X error on the target qubit	1(a), 2, or 4(a)
Orange CNOT fault with Y or Z error on the target qubit	2 or 4(a)

A list of possible faults during the flag-FT operator measurement protocol and corresponding correction procedures is given in Table II.

The flag-FT measurement protocol described above is for a measurement of an operator commuting with all generators which acts in one code block. Surprisingly, the protocol also works for an operator acting on two or more code blocks. The measurement of such operator can be done by treating parts of the operator acting on different code blocks as operators from different subblocks. For example, let C_p and C_q be cyclic CSS codes of distance 3 satisfying Theorem 2, and let P and Q be Pauli operators acting on C_p and C_q, respectively. The measurement of $P \otimes Q$ on the code $C_p \otimes C_q$ can be done by

using a one-flag circuit given in Fig. 3, where P and Q are treated as operators from the different subblocks. Observe that if f_0 flags and at least one f_i flags [the 4(b) or 4(c) case], the resulting error after appropriate operation will become a consecutive error on either first or second code blocks. Since C_p and C_q are both cyclic, we can determine the error by performing subsequent syndrome measurement on only C_p or C_q, depending on the subblock in which the fault occurs. After that, the correct measurement result can be obtained by a subsequent operator measurement.

VI. APPLICATIONS OF FAULT-TOLERANT OPERATOR MEASUREMENT PROTOCOL

A measurement of an operator commuting with all generators can be used as a subroutine in numerous quantum information processing techniques such as state preparation and quantum gate implementation. Since the fault-tolerant measurement protocol described in Sec. V is applicable on two or more code blocks, information processing between code blocks is possible. In this section we briefly describe some important techniques which make fault-tolerant computation on cyclic CSS codes possible, including logical Einstein-Podolsky-Rosen (EPR) state preparation, teleportation, and quantum computation on logical qubits. Readers who are familiar with Clifford gate implementations via quantum gate teleportation may skip this section. The purpose of this section is to justify that all related techniques can be done in a fault-tolerant way on logical qubits using our measurement protocol.

Let us consider an EPR state $\frac{|00\rangle+|11\rangle}{\sqrt{2}}$. This is a $+1$ eigenstate of operators $X\otimes X$ and $Z\otimes Z$. Let C_p and C_q be $[\![n_1,k_1,d_1]\!]$ and $[\![n_2,k_2,d_2]\!]$ cyclic CSS codes satisfying Theorem 2 with stabilizer generating sets $\{g^p_{i_1}\}$ and $\{g^q_{i_2}\}$, respectively. Suppose that we want to prepare a state

$$\frac{|\bar{0}\rangle_{p,i}|\bar{0}\rangle_{q,j}+|\bar{1}\rangle_{p,i}|\bar{1}\rangle_{q,j}}{\sqrt{2}}, \tag{22}$$

which is an EPR state between the ith logical qubit of C_p and the jth logical qubit of C_q. This can be done by performing projective measurements with respect to stabilizer generators $\{g^p_{i_1}\otimes I, I\otimes g^q_{i_2}\}$ and logical operators $\bar{X}_{p,i}\otimes\bar{X}_{q,j}$ and $\bar{Z}_{p,i}\otimes\bar{Z}_{q,j}$ on a totally mixed state, where $\bar{X}_{p,i}$ and $\bar{X}_{q,j}$ (or $\bar{Z}_{p,i}$ and $\bar{Z}_{q,j}$) are logical X (or logical Z) operators on ith logical qubit of C_p and jth logical qubit of C_q, respectively. Since the measurement protocol described in Sec. V is a fault-tolerant protocol, the state in Eq. (22) can be prepared in a fault-tolerant way.

In conventional quantum teleportation, an EPR state and Bell measurement are required. Here we will examine a process for fault-tolerant quantum teleportation of logical data between two code blocks. The scheme for logical qubit teleportation is shown in Fig. 5. Suppose that we would like to teleport the ith logical qubit of C_p to the jth logical qubit of C_q, first an EPR state $\frac{|\bar{0}\rangle_{q,j}|\bar{0}\rangle_{q,j}+|\bar{1}\rangle_{q,j}|\bar{1}\rangle_{q,j}}{\sqrt{2}}$ prepared on $C_q\otimes C_q$ is required. The logical qubit teleportation can be done by performing a Bell measurement with respect to $\bar{X}_{p,i}\otimes\bar{X}_{q,j}$ and $\bar{Z}_{p,i}\otimes\bar{Z}_{q,j}$ between C_p and the first block of C_q. The teleported logical qubit can be obtained in the second

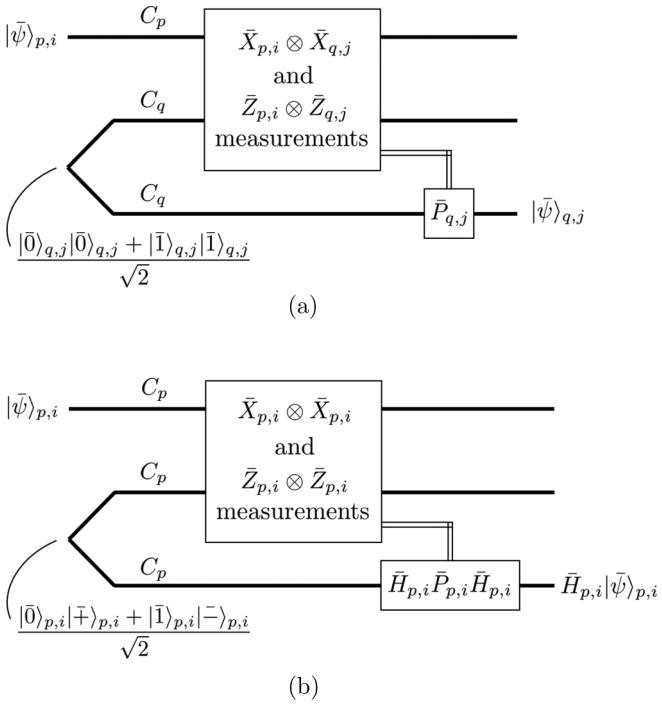

FIG. 5. Schemes for teleportation and Clifford gate implementation on cyclic CSS codes. A bold line represents a block of code, while a double line represents classical information. (a) The ith logical qubit of C_p is teleported to the jth logical qubit of C_q. (b) A logical Hadamard gate $\bar{H}_i$ is performed on the ith logical qubit of C_p via quantum gate teleportation.

block of C_q by operating an appropriate logical Pauli operator $\bar{P}_{q,j}$ depending on the Bell measurement result. Note that the Bell measurement can be done fault-tolerantly using the measurement protocol in Sec. V, and logical Pauli operators are transversal (therefore, fault-tolerant). Thus, fault-tolerant teleportation between two code blocks can be achieved.

Now let us consider fault-tolerant computation on cyclic CSS codes. It is known that for any error correcting code, by the Eastin-Knill theorem [29], at least one logical gate in a universal gate set cannot be implemented transversely. For such gates, other fault-tolerant techniques must be performed, which can require a significant amount of resources. Fortunately, fault-tolerant implementations of logical Clifford gates on distance-3 cyclic CSS codes can be achieved via quantum gate teleportation (see Ref. [30] for the details of quantum gate teleportation). For example, suppose that we would like to perform a logical Hadamard gate $\bar{H}_i$ on the code C_p. This can be achieved by preparing a codeword which is an eigenstate of $\bar{X}_i\otimes\bar{Z}_i$ and $\bar{Z}_i\otimes\bar{X}_i$ on $C_p\otimes C_p$, performing logical qubit teleportation, and operating a logical Pauli operator $\bar{H}_{p,i}\bar{P}_{p,i}\bar{H}_{p,i}$ depending on the result from the Bell measurement in qubit teleportation as illustrated in Fig. 5(b). Also, logical $R_{\pi/4}=\mathrm{diag}(1,\mathrm{i})$ and logical CNOT gates on any logical qubits can be performed in a similar way. The Clifford group can be generated by $\{H, R_{\pi/4}, \mathrm{CNOT}\}$ [31,32]. Thus, our scheme is applicable to any Clifford operation. It is known that universal quantum computation can be achieved by Clifford gates and any other gate not in the Clifford group [33]. However, performing logical non-Clifford gates will require different techniques such as the ones presented in Ref. [20].

TABLE III. A choice of logical operators for the $[\![30, 14, 3]\!]$ code.

$\bar{X}_1$	$X_1X_{11}X_{21}$	$\bar{Z}_1$	$Z_1Z_{11}Z_{21}$
$\bar{X}_2$	$X_2X_{12}X_{22}$	$\bar{Z}_2$	$Z_2Z_{12}Z_{22}$
$\bar{X}_3$	$X_3X_{13}X_{23}$	$\bar{Z}_3$	$Z_3Z_{13}Z_{23}$
$\bar{X}_4$	$X_4X_{14}X_{24}$	$\bar{Z}_4$	$Z_4Z_{14}Z_{24}$
$\bar{X}_5$	$X_5X_{15}X_{25}$	$\bar{Z}_5$	$Z_5Z_{15}Z_{25}$
$\bar{X}_6$	$X_6X_{16}X_{26}$	$\bar{Z}_6$	$Z_6Z_{16}Z_{26}$
$\bar{X}_7$	$X_7X_{17}X_{27}$	$\bar{Z}_7$	$Z_7Z_{17}Z_{27}$
$\bar{X}_8$	$X_8X_{18}X_{28}$	$\bar{Z}_8$	$Z_8Z_{18}Z_{28}$
$\bar{X}_9$	$X_9X_{19}X_{29}$	$\bar{Z}_9$	$Z_9Z_{19}Z_{29}$
$\bar{X}_{10}$	$X_{10}X_{20}X_{30}$	$\bar{Z}_{10}$	$Z_{10}Z_{20}Z_{30}$
$\bar{X}_{11}$	$X_1X_7X_9X_{11}X_{17}X_{19}$	$\bar{Z}_{11}$	$Z_{11}Z_{17}Z_{19}Z_{21}Z_{27}Z_{29}$
$\bar{X}_{12}$	$X_2X_8X_{10}X_{12}X_{18}X_{20}$	$\bar{Z}_{12}$	$Z_{12}Z_{18}Z_{20}Z_{22}Z_{28}Z_{30}$
$\bar{X}_{13}$	$X_{11}X_{17}X_{19}X_{21}X_{27}X_{29}$	$\bar{Z}_{13}$	$Z_1Z_7Z_9Z_{11}Z_{17}Z_{19}$
$\bar{X}_{14}$	$X_{12}X_{18}X_{20}X_{22}X_{28}X_{30}$	$\bar{Z}_{14}$	$Z_2Z_8Z_{10}Z_{12}Z_{18}Z_{20}$

VII. EXAMPLES OF CYCLIC CSS CODES

In this section, some examples of cyclic CSS codes satisfying the error-distinguishability theorem (Theorem 2) are given. A first example is the $[\![7, 1, 3]\!]$ quantum Hamming code. This code is constructed from a classical [7,4,3] Hamming code (with $C_x = C_z$). A check polynomial of the [7,4,3] Hamming code in cyclic form is

$$h(x) = 1 + x^2 + x^3 + x^4. \tag{23}$$

In fact, any classical Hamming code can be made cyclic [22]. Thus, any CSS code constructed from a classical $[2^r - 1, 2^r - 1 - r, 3]$ Hamming code with $C_x = C_z$ satisfies Theorem 2, and can be used in the flag-FTEC protocol and the flag-FT measurement protocol described in this work.

Another example of cyclic CSS codes satisfying Theorem 2 is the $[\![30, 14, 3]\!]$ code constructed from a classical [30,22,3] cyclic code with a check polynomial

$$h(x) = 1 + x^2 + x^4 + x^6 + x^{10} + x^{14} + x^{16} + x^{22}. \tag{24}$$

The [30,22,3] code and other classical codes satisfying $C^\perp \subseteq C$ are given in Table 1 of Ref. [28]. (A method of finding the check polynomial of a classical cyclic code is discussed in Ref. [22]). One possible choice of logical operators for the $[\![30, 14, 3]\!]$ code is given in Table III. The advantages of the $[\![30, 14, 3]\!]$ code are that its encoding rate is high ($k/n = 14/30$), and the logical operators of the first 10 logical qubits have a simple form, which make them easily accessible.

VIII. DISCUSSION AND CONCLUSION

In this work we used the symmetries of CSS codes constructed from classical cyclic codes to prove that errors written in consecutive form (as in Definition 7) can be distinguished. From these properties, we can obtain a one-flag circuit along with a flag-FTEC protocol which satisfies the criteria of FTEC in Definition 2 when there is at most one fault. The one-flag circuit requires only four ancilla qubits. This number does not grow as the block length gets larger, making our protocol advantageous in the implementation where resources are limited. We note that not all cyclic CSS codes are Hamming codes and therefore the methods in Ref. [13] (which apply to perfect codes) cannot be directly applied, thus providing further motivation for our work.

In general, cyclic CSS codes do not satisfy the sufficient condition required for flag fault-tolerance presented in Ref. [16] (one example is the family of Hamming codes which can be made cyclic). Nevertheless, using the techniques presented in this paper, a flag-FTEC protocol can still be achieved.

Furthermore, we have shown how logical Pauli operators of cyclic CSS codes can be measured in a fault-tolerant way using the flag techniques discussed in Sec. V. The flag-FT operator measurement protocol satisfies the criteria of FT nondestructive measurement in Definition 10 when there is at most one fault. We then showed in Sec. VI how one can perform quantum gate teleportation in a fault-tolerant way to implement logical Clifford operators on any given logical qubit for codes which encode multiple logical qubits. Examples of cyclic CSS codes with large encoding rates are provided in Sec. VII.

Note that for all CSS codes, the stabilizer generators being measured are of the form $I^{\otimes n-m} \otimes X^{\otimes m}$ or $I^{\otimes n-m} \otimes Z^{\otimes m}$ up to qubit permutations. Thus data qubit errors arising from faulty CNOT gates will be expressed in consecutive form. The errors of this form are distinguishable iff the submatrices of the X and Z stabilizers satisfy the consecutive error lemma for general CSS codes (Lemma 1). In our work, we use the symmetry of the cyclic codes to simplify Lemma 1 and obtain the consecutive error lemma for cyclic CSS codes (Lemma 3). We believe that Lemma 1 can be simplified by using symmetries found in other families of quantum codes. With appropriate t-flag circuits and operations depending on the flag measurement outcome, this may lead to new flag fault-tolerant protocols.

Another interesting avenue is finding noncyclic quantum codes for which a version of the error-distinguishability theorem (Theorem 2) can be applied. We note that for such codes, the same one-flag circuit as in Fig. 3 along with the flag-FTEC protocol of Sec. IV and the flag-FT measurement protocol of Sec. V can be used. The reason is that the key property used by these schemes is based on the distinguishability of consecutive errors.

Note that there are quantum cyclic codes which are not CSS codes for which flag fault-tolerant schemes are still possible. For instance, a flag-FTEC protocol for the $[\![5, 1, 3]\!]$ code was devised in Ref. [13]. We believe that it could be interesting to generalize the ideas presented in this work to non-CSS cyclic quantum codes. However, we leave this problem for future work.

The flag fault-tolerant protocols for cyclic CSS codes presented in this work are based on the assumption that the qubit measurement and state preparation must be fast since we reuse some flag qubits in the protocols (as we can see in Fig. 3). If we do not reuse flag qubits, however, the number of required ancillas will be $m + 2$ for an operator being measured of the form $P = P_1^{\otimes a_1} \otimes I^{\otimes b_1} \otimes P_2^{\otimes a_2} \otimes I^{\otimes b_2} \otimes \cdots \otimes P_m^{\otimes a_m} \otimes I^{\otimes b_m}$ instead of 4.

One important feature of flag fault-tolerant protocols is that the number of required ancillas is very small compared to other fault-tolerant schemes. We believe that if fewer ancillas are required, the accuracy threshold will increase since the

number of locations will decrease in total. However, we should point out that subsequent syndrome measurements are also required in a flag fault-tolerant protocol and may increase the total number of locations in the protocol. The answer of whether the accuracy threshold for a flag fault-tolerant protocol is greater or smaller compared to other fault-tolerant schemes when a cyclic CSS code is being used is still unknown. An example of simulations to obtain thresholds for flag error correction using other code families can be found in Ref. [16].

Last, we point out that cyclic CSS codes which satisfy the condition in Theorem 2 are not limited to distance-3 codes. Therefore, interesting future work would be to use the methods of Ref. [16] to obtain flag fault-tolerant schemes for higher-distance codes. In particular, the main challenge stems from finding t-flag circuits as in Fig. 3 for $t > 1$.

ACKNOWLEDGMENTS

T.T. acknowledges the support of The Queen Sirikit Scholarship under The Royal Patronage of Her Majesty Queen Sirikit of Thailand. C.C. acknowledges the support of NSERC through the PGS D scholarship. D.L. is supported by an NSERC Discovery grant and a CIFAR research grant via the Quantum Information Science program. Perimeter Institute is supported in part by the Government of Canada and the Province of Ontario.

APPENDIX: PROOF OF THE LEMMAS

Proof of Lemma 1. We will prove that $\mathcal{E}^z_{0,n}$ is distinguishable by C iff for all $p, q \in \{0, 1, ..., n-1\}$ such that $p > q$, there exists $i \in \{1, \ldots, r_x\}$ such that $x_{i,n-p+1} \oplus \cdots \oplus x_{i,n-q} = 1$. Consider errors $E_p = I^{\otimes n-p} \otimes Z^{\otimes p}$ and $E_q = I^{\otimes n-q} \otimes Z^{\otimes q}$ where $p, q \in \{0, 1, \ldots, n-1\}$, $p > q$. Let $s(E_p), s(E_q) \in \mathbb{Z}_2^r$ be error syndromes corresponding to errors E_p and E_q, respectively. By the definition of distinguishable errors (Definition 5), E_p and E_q are distinguishable by C iff $s(E_p) \neq s(E_q)$, i.e., there exists $i \in \{1, 2, \ldots, r_x\}$ such that $s(E_p)_i \neq s(E_q)_i$ [here i corresponds to the ith component of $s(E_p)$ and $S(E_q)$]. From the parity check matrix H_x, the ith component of $s(E_p)$ and $s(E_q)$ is given by

$$s(E_p)_i = x_{i,n-p+1} \oplus x_{i,n-p+2} \oplus \cdots \oplus x_{i,n}, \tag{A1}$$

$$s(E_q)_i = x_{i,n-q+1} \oplus x_{i,n-q+2} \oplus \cdots \oplus x_{i,n}. \tag{A2}$$

From Eqs. (A1) and (A2), we have that

$$\begin{aligned} s(E_p)_i \neq s(E_q)_i &\Leftrightarrow s(E_p)_i \oplus s(E_q)_i = 1 \\ &\Leftrightarrow x_{i,n-p+1} \oplus \cdots \oplus x_{i,n-q} = 1. \end{aligned} \tag{A3}$$

Thus, $\mathcal{E}^z_{0,n}$ is distinguishable by C iff for all $p, q \in \{0, 1, \ldots, n-1\}$ such that $p > q$, there exists $i \in \{1, 2, \ldots, r_x\}$ such that

$$x_{i,n-p+1} \oplus \cdots \oplus x_{i,n-q} = 1. \tag{A4}$$

The proof of the statement for $\mathcal{E}^x_{0,n}$ is similar.

Now we will proof that $\mathcal{E}^P_{0,n}$ is distinguishable by C iff both $\mathcal{E}^z_{0,n}$ and $\mathcal{E}^x_{0,n}$ are distinguishable by C. Let $X_p = I^{\otimes n-p} \otimes X^{\otimes p}$ and $Z_q = I^{\otimes n-q} \otimes Z^{\otimes q}$, where $p, q \in \{0, \ldots, n-1\}$. Observe that any element of $\mathcal{E}^P_{0,n}$ is of the form $E_{p,q} = X_p \cdot Z_q$ where $X_p \in \mathcal{E}^x_{0,n}$ and $Z_q \in \mathcal{E}^z_{0,n}$. The syndrome of $E_{p,q}$ is $s(E_{p,q}) = (s(X_p)|s(Z_q))$. If $\mathcal{E}^P_{0,n}$ is distinguishable by C, i.e., $s(E_{p_1,q_1}) \neq s(E_{p_2,q_2})$ for all choices of p_1, p_2, q_1, q_2 such that $(p_1, q_1) \neq (p_2, q_2)$, then we have that any pair of X_{p_1} and X_{p_2} and any pair of Z_{q_1} and Z_{q_2} are distinguishable. Conversely, if any pair of X_{p_1} and X_{p_2} and any pair of Z_{q_1} and Z_{q_2} are distinguishable, then any pair of E_{p_1,q_1} and E_{p_2,q_2} will have different syndromes. This implies statement 3.

Proof of Lemma 2. Suppose that the stabilizer group of C can be generated by $\{g_1, g_2, \ldots, g_{n-k}\}$. Since C is a CSS code, we will first assume that the generators g_i's are either X type or Z type, denoted as g_i^x or g_i^z. Let H_x and H_z be X and Z stabilizer matrices of the code C in symplectic representation, and let $C_x^\perp$ and $C_z^\perp$ be the classical codes generated by H_x and H_z, respectively. Observe that any element of C in symplectic representation is of the form $(x|z)$ where $x \in C_x^\perp$ and $z \in C_z^\perp$. For any choice of $l \in \{0, 1, \ldots, n-1\}$, let $\tilde{H}_x$ (or $\tilde{H}_z$) be the parity check matrix corresponding to $\mathcal{L}(g_i^x, l)$'s (or $\mathcal{L}(g_i^z, l)$'s). We find that the code $\tilde{C}_x^\perp$ generated by $\tilde{H}_x$ (or $\tilde{C}_z^\perp$ generated by $\tilde{H}_z$) differs from $C_x^\perp$ (or $C_z^\perp$) by an l-step left cyclic permutation. However, since $C_x^\perp$ and $C_z^\perp$ are cyclic codes, we have that $\tilde{C}_x^\perp = C_x^\perp$ and $\tilde{C}_z^\perp = C_z^\perp$. Therefore, $\{\mathcal{L}(g_1^x, l), \ldots, \mathcal{L}(g_{r_x}^x, l), \mathcal{L}(g_1^z, l), \ldots, \mathcal{L}(g_{r_z}^z, l)\}$ and $\{g_1^x, \ldots, g_{r_x}^x, g_1^z, \ldots, g_{r_z}^z\}$ generate the same stabilizer group for any $l \in \{0, 1, \ldots, n-1\}$.

In general, some generators of the stabilizer group of C might be neither X type nor Z type. The following transformations of the generators preserve the stabilizer group, and the last set of generators is the cyclic shifts of the original: (1) Transform the given generators to either X type or Z type. This corresponds to appropriate reversible row operations on the binary symplectic representation of C to obtain the block diagonal form,

$$\left(\begin{array}{c|c} A & 0 \\ 0 & B \end{array}\right). \tag{A5}$$

(2) Cyclic shifts of these resulting generators also generate the same stabilizer group. (3) Reversing the transformation in step (1) [now applied to the generators after step (2)] preserves the stabilizer group. The resulting generators are cyclic shifts of the original.

Proof of Lemma 3. First we will prove that $\mathcal{E}^z_{0,n}$ is distinguishable by C iff for all $u_x \in \{2, 3, \ldots, n\}$, there exists $i \in \{1, \ldots, r_x\}$ such that $x_{i,u_x} \oplus \cdots \oplus x_{i,n} = 1$. Applying the consecutive error lemma for general CSS codes (Lemma 1), we would like to prove that for all $p, q \in \{0, \ldots, n-1\}$ such that $p > q$, there exists $i \in \{1, \ldots, r_x\}$ such that $x_{i,n-p+1} \oplus \cdots \oplus x_{i,n-q} = 1$ iff for all $u_x \in \{2, 3, \ldots, n\}$, there exist $i' \in \{1, \ldots, r_x\}$ such that $x_{i',u_x} \oplus \cdots \oplus x_{i',n} = 1$.

($\Rightarrow$) By choosing $q = 0$ and $p = n - u + 1$, the proof is trivial.

($\Leftarrow$) Assume by contradiction that there exists a pair of $p, q \in \{0, 1, \ldots, n-1\}$ with $p > q$ such that $x_{i,n-p+1} \oplus \cdots \oplus x_{i,n-q} = 0$ for all i; i.e., there exists a pair of errors $E_p = I^{\otimes n-p} \otimes Z^{\otimes p}$ and $E_q = I^{\otimes n-q} \otimes Z^{\otimes q}$ which cannot be distinguished by any generator of C. Let C be generated by

$\{g_1, \ldots, g_r\}$. By the cyclic permutation lemma (Lemma 2), we can construct a different generator set $\{\tilde{g}_1, \ldots, \tilde{g}_r\}$ of C where $\tilde{g}_i = \mathcal{L}(g_i, q)$ for all i. Let the X part of $\sigma(\tilde{g}_i)$ be $(\tilde{x}_{i,1}, \ldots, \tilde{x}_{i,n}) = (x_{i,q+1}, \ldots, x_{i,n}, x_{i,1}, \ldots, x_{i,q})$. Note that $\tilde{x}_{n-p+1} = x_{i,n-(p-q)+1}$ and $\tilde{x}_{n-q} = x_{i,n}$. The assumption implies that E_p and E_q cannot be distinguished by any $\tilde{g}_i$ as well. This gives

$$\tilde{x}_{i,n-p+1} \oplus \cdots \oplus \tilde{x}_{i,n-q} = 0, \tag{A6}$$

or equivalently,

$$x_{i,n-(p-q)+1} \oplus \cdots \oplus x_{i,n} = 0. \tag{A7}$$

Let $u_x = n - (p - q) + 1$. Therefore, there exists $u_x \in \{2, 3, \ldots, n\}$ such that $x_{i,u_x} \oplus \cdots \oplus x_{i,n} = 0$ for all i.

The proof of statement for $\mathcal{E}^x_{0,n}$ is similar to the proof of statement for $\mathcal{E}^z_{0,n}$, while the proof of statement for $\mathcal{E}^P_{0,n}$ is similar to the proof of statement 3 in Lemma 1.

We already proved statements for $\mathcal{E}^z_{0,n}$, $\mathcal{E}^x_{0,n}$, and $\mathcal{E}^P_{0,n}$. We will generalize the statements to $\mathcal{E}^z_{l,n}$, $\mathcal{E}^x_{l,n}$, and $\mathcal{E}^P_{l,n}$ for any $l \in \{0, \ldots, n-1\}$. Let $\tilde{C}$ by a cyclic CSS code generated by $\{\mathcal{L}(g^x_1, l), \ldots, \mathcal{L}(g^x_{r_x}, l), \mathcal{L}(g^z_1, l), \ldots, \mathcal{L}(g^z_{r_z}, l)\}$. Observe that by qubit reordering, $\mathcal{E}^P_{l,n}$ is distinguishable by $\tilde{C}$ iff $\mathcal{E}^P_{0,n}$ is distinguishable by C. Since $\tilde{C}$ and C are the same code by the cyclic permutation lemma (Lemma 2), we have that $\mathcal{E}^P_{l,n}$ is distinguishable by C for any $l \in \{0, \ldots, n-1\}$ iff $\mathcal{E}^P_{0,n}$ is distinguishable by C. The proof is also applied to $\mathcal{E}^z_{l,n}$ and $\mathcal{E}^x_{l,n}$.

[1] P. W. Shor, Fault-tolerant quantum computation, in *Proceedings of the 37th Annual Symposium on Foundations of Computer Science* (IEEE Computer Society Press, Los Alamitos, California, 1996), pp. 56–65.

[2] D. Aharonov and M. Ben-Or, Fault-tolerant quantum computation with constant error, in *Proceedings of the Twenty-ninth Annual ACM Symposium on Theory of Computing* (ACM, New York, 1997), pp. 176–188.

[3] J. Preskill, Reliable quantum computers, Proc. R. Soc. London A **454**, 385 (1998).

[4] E. Knill, R. Laflamme, and W. H. Zurek, Threshold accuracy for quantum computation, arXiv:quant-ph/9610011 (1996).

[5] E. Dennis, A. Kitaev, A. Landhal, and J. Preskill, Topological quantum memory, J. Math. Phys. **43**, 4452 (2002).

[6] R. Raussendorf and J. Harrington, Fault-Tolerant Quantum Computation with High Threshold in Two Dimensions, Phys. Rev. Lett. **98**, 190504 (2007).

[7] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, Phys. Rev. A **86**, 032324 (2012).

[8] Y. Tomita and K. M. Svore, Low-distance surface codes under realistic quantum noise, Phys. Rev. A **90**, 062320 (2014).

[9] A. G. Fowler, A. C. Whiteside, A. L. McInnes, and A. Rabbani, Topological Code Autotune, Phys. Rev. X **2**, 041003 (2012).

[10] A. Paetznick and B. W. Reichardt, Fault-tolerant ancilla preparation and noise threshold lower bounds for the 23-qubit Golay code, Quantum Info. Comput. **12**, 1034 (2012).

[11] C. Chamberland, T. Jochym-O'Connor, and R. Laflamme, Overhead analysis of universal concatenated quantum codes, Phys. Rev. A **95**, 022313 (2017).

[12] R. Takagi, T. J. Yoder, and I. L. Chuang, Error rates and resource overheads of encoded three-qubit gates, Phys. Rev. A **96**, 042302 (2017).

[13] R. Chao and B. W. Reichardt, Quantum Error Correction with Only Two Extra Qubits, Phys. Rev. Lett. **121**, 050502 (2018).

[14] R. Chao and B. W. Reichardt, Fault-tolerant quantum computation with few qubits, npj Quantum Inf. **4**, 42 (2018).

[15] B. W. Reichardt, Fault-tolerant quantum error correction for Steane's seven-qubit color code with few or no extra qubits, arXiv:1804.06995 (2018).

[16] C. Chamberland and M. E. Beverland, Flag fault-tolerant error correction with arbitrary distance codes, Quantum **2**, 53 (2018).

[17] A. A. Kovalev and L. P. Pryadko, Fault tolerance of quantum low-density parity check codes with sublinear distance scaling, Phys. Rev. A **87**, 020304(R) (2013).

[18] D. Gottesman, Fault-tolerant quantum computation with constant overhead, Quantum Inf. Comput. **14**, 1338 (2014).

[19] J. P. Tillich and G. Zémor, Quantum LDPC codes with positive rate and minimum distance proportional to the square root of the blocklength, IEEE Trans. Inf. Theory **60**, 1193 (2014).

[20] C. Chamberland and A. W. Cross, Fault-tolerant magic state preparation with flag qubits, Quantum **3**, 143 (2019).

[21] Y. Shi, C. Chamberland, and A. W. Cross, Fault-tolerant preparation of approximate GKP states, New J. Phys. **21**, 093007 (2019).

[22] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes* (North-Holland, New York, 1977).

[23] D. Gottesman, Class of quantum error-correcting codes saturating the quantum Hamming bound, Phys. Rev. A **54**, 1862 (1996).

[24] D. Gottesman, Stabilizer codes and quantum error correction, Ph.D. thesis, California Institute of Technology, 1997.

[25] A. R. Calderbank and P. W. Shor, Good quantum error-correcting codes exist, Phys. Rev. A **54**, 1098 (1996).

[26] A. W. Steane, Multiple-particle interference and quantum error correction, Proc. R. Soc. London **452**, 2551 (1996).

[27] P. Aliferis, D. Gottesman, and J. Preskill, Quantum accuracy threshold for concatenated distance-3 codes, Quantum Inf. Comput. **6**, 97 (2006).

[28] R. Li and X. Li, Quantum codes constructed from binary cyclic codes, Intl. J. Quantum Inform. **02**, 265 (2004).

[29] B. Eastin and E. Knill, Restrictions on Transversal Encoded Quantum Gate Sets, Phys. Rev. Lett. **102**, 110502 (2009).

[30] D. Gottesman and I. L. Chuang, Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations, Nature (London) **402**, 390 (1999).

[31] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, Quantum Error Correction and Orthogonal Geometry, Phys. Rev. Lett. **78**, 405 (1997).

[32] D. Gottesman, Theory of fault-tolerant quantum computation, Phys. Rev. A **57**, 127 (1998).

[33] G. Nebe, E. M. Rains, and N. J. Sloane, The invariants of the Clifford groups, Designs Codes Cryptogr. **24**, 99 (2001).