nature portfolio

https://doi.org/10.1038/s41586-026-10709-y

Supplementary information

Universal gates from braiding and fusing anyons on quantum hardware

In the format provided by the authors and unedited

Supplementary Information

Note: the section labels (with Roman numerals) refer to sections in the main text as follows:

S1. BASICS OF S_3 QUANTUM DOUBLE MODEL

For more details, we refer the reader to the comprehensive pedagogical notes in Refs. S1 and S2.

A. Review of S_3 group theory and representation theory

We use the group presentation for $S_3 = \langle \mu, \sigma | \mu^3 = \sigma^2 = e, \mu\sigma = \sigma\mu^{-1} \rangle$, where μ and σ corresponds to the 120° rotation and reflection of an equilateral triangle. Another common notation is to recognize that S_3 is a symmetric group on three elements, so group elements are permutations that can be represented in terms of cycles. The identification is $\mu = (123)$, $\sigma = (23)$ (the remaining group elements can be determined from these two generators).

There are three conjugacy classes that partitions S_3: $C_1 = \{ e \}$, $C_2 = \{ \sigma, \mu\sigma, \bar{\mu}\sigma \}$, $C_3 = \{ \mu, \bar{\mu} \}$.

There are three irreducible representations (irreps) of S_3: $[+], [-], [2]$. The trivial representation $[+]$ sends every group element to 1. The sign representation $[-]$ sends group element $g \mapsto \mathrm{sgn}(g)$, where sgn gives the sign of the permutation; so even permutations (e.g. C_3 group elements) are mapped to $+1$, whereas odd permutations (e.g. C_2 group elements) are mapped to -1. The standard representation $[2]$ is a two-dimensional representation, and the representation matrices are:

$$\Gamma_{[2]}(e) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \Gamma_{[2]}(\mu) = \begin{pmatrix} \omega & 0 \\ 0 & \bar{\omega} \end{pmatrix} \qquad \Gamma_{[2]}(\bar{\mu}) = \begin{pmatrix} \bar{\omega} & 0 \\ 0 & \omega \end{pmatrix}$$
$$\Gamma_{[2]}(\mu\sigma) = \begin{pmatrix} 0 & \omega \\ \bar{\omega} & 0 \end{pmatrix} \qquad \Gamma_{[2]}(\bar{\mu}\sigma) = \begin{pmatrix} 0 & \bar{\omega} \\ \omega & 0 \end{pmatrix} \qquad \Gamma_{[2]}(\sigma) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

where the indexing order is $\{|2+\rangle, |2-\rangle\}$ and $\omega = \exp(2\pi i/3)$.

The representation theory of the $\mathbb{Z}_3$ subgroup of S_3 also warrants a mention, as it will be important for defining some of the anyon types in S_3 quantum double. Since $\mathbb{Z}_3$ is an Abelian group, all irreps are one-dimensional. The three possible irreps are labeled by the third root of unity. The trivial representation $[1]$ maps all group elements to 1. The $[\omega]$ representation is given by the map $a \mapsto \omega^a$. The $[\bar{\omega}]$ representation is given by the map $a \mapsto (\bar{\omega})^a$.

B. Qutrit-qubit encoding

We use the qutrit-qubit encoding for the S_3 group elements, such that for a given S_3 group element basis $|g\rangle = |\mu^a \sigma^b\rangle$, it is encoded as a qutrit $|a \pmod 3\rangle$ and a qubit $|b \pmod 2\rangle$, where a, b are integers.

The left- and right-group multiplication operators are

$$L^g |h\rangle = |gh\rangle , \qquad R^g |h\rangle = |h\bar{g}\rangle , \tag{S1}$$

where $\bar{g}$ is the inverse of g. In terms of qutrit and qubit gates, the left- and right- group multiplication of S_3 generators are

$$L^{\mu} = \mathcal{X} \otimes I \,, \quad L^{\sigma} = \mathcal{C} \otimes X \,, \tag{S2}$$

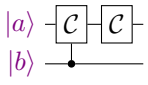
$$R^{\mu} = \mathcal{X}^{-Z} \,, \quad R^{\sigma} = I \otimes X \,, \tag{S3}$$

where $\mathcal{X}$ and $\mathcal{Z}$ denote the qutrit clock matrices, and $\mathcal{C}$ is the qutrit charge-conjugation operator,

$$\mathcal{Z} = \begin{pmatrix} 1 & & \\ & \omega & \\ & & \omega^2 \end{pmatrix}, \quad \mathcal{X} = \begin{pmatrix} & & 1 \\ 1 & & \\ & 1 & \end{pmatrix}, \quad \mathcal{C} = \begin{pmatrix} 1 & & \\ & & 1 \\ & 1 & \end{pmatrix} , \tag{S4}$$

and where Z and X are the Pauli matrices acting on the qubit degrees of freedom.

Given an S_3 group element $g = \mu^a \sigma^b$, its inverse is $\bar{g} = \mu^{-(-1)^b a} \sigma^b$. This inverse operation is implemented by the following circuit:

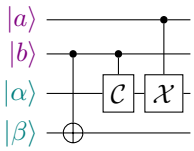


(S5)

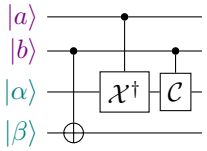
Note that taking an inverse of a S_3 group element does not change the qubit value b since inverse operation does not change its conjugacy class.

Controlled left multiplication of a group element $h = \mu^{\alpha} \sigma^{\beta}$ (in teal) by the group element $g = \mu^a \sigma^b$ (in violet) (left panel) and its inverse $\bar{g}$ (right panel) is

$|g\rangle\,|h\rangle \to |g\rangle\,|gh\rangle$:



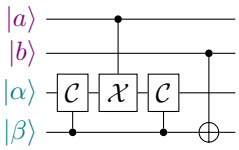
$|g\rangle\,|h\rangle \to |g\rangle\,|\bar{g}h\rangle$:



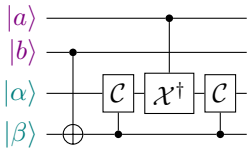
(S6)

Similarly, controlled right multiplication by a group element g (left panel) and its inverse $\bar{g}$ (right panel) is

$|g\rangle\,|h\rangle \to |g\rangle\,|hg\rangle$:

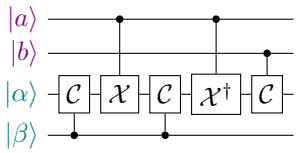


$|g\rangle\,|h\rangle \to |g\rangle\,|h\bar{g}\rangle$:



(S7)

Finally, controlled conjugation sends $g \to \bar{z} g z$, where $z = \mu^a \sigma^b$ is the control (in violet) and $g = \mu^{\alpha} \sigma^{\beta}$ is the target (in teal). So the circuit is



(S8)

C. Anyon types, fusion rules, and braiding

Given a finite group G, the quantum double of G (also known as the G Drinfeld double, denoted as $\mathcal{D}(G)$) [S3], has anyon type specified by (C, χ), where C is a conjugacy class and χ is an irreducible representation of the centralizer $Z(r)$ for a representative r in C. The 8 possible anyon types in S_3 quantum double are given below:

We denote the vacuum anyon $(C_1, [+])$ by 1 in the main text. In the following, we also simply refer to it as $[+]$.

The most relevant fusion rules are as follows:

$$[-] \times [-] = [+], \quad [2] \times [2] = [+] + [-] + [2] \tag{S9}$$

Conjugacy class	Centralizer	Charge	d
C_1	$Z(e) = S_3$	$[+]$	1
		$[-]$	1
		$[2]$	2
C_2	$Z(\sigma) \cong \mathbb{Z}_2$	$[+]$	3
		$[-]$	3
C_3	$Z(\mu) \cong \mathbb{Z}_3$	$[1]$	2
		$[\omega]$	2
		$[\bar{\omega}]$	2

Table I. Conjugacy classes, centralizers, charges, and the quantum dimensions (d) of the 8 anyon types in $\mathcal{D}(S_3)$.

$$C_2 \times C_2 = [+] + [2] + C_3 + (C_3, [\omega]) + (C_3, [\bar{\omega}]), \quad C_2 \times [2] = C_2 + (C_2, [-]) \tag{S10}$$

$$C_3 \times C_3 = [+] + [-] + C_3, \quad C_3 \times C_2 = C_2 + (C_2, [-]) \tag{S11}$$

The full categorical data describing the S_3 quantum double, including fusion rules, the F symbols, and the R matrices, can be found in Ref. S4 and in the Supplementary Categorical Data of Ref. S5.

For braiding, it suffices to describe two scenarios: 1) braiding two fluxes and 2) braiding a flux around a charge. Braiding involving dyons can be decomposed into aforementioned scenarios. For scenario 1, when two fluxes are braided around each other, the global fluxes labels are both conjugated by the total flux. For scenario 2, the action of a flux g on a charge's internal Hilbert space is given by the representation matrix $\Gamma(g)$, where Γ is the irreducible representation of the charge.

D. Ribbon operators

1. *Basic construction*

The operators that create pairs of non-Abelian anyons on top of the quantum double ground state are known as *ribbon operators.* A ribbon operator $F_\rho^{A;u,u'}$ is specified by four data: (i) the path along the lattice where the ribbon acts, ρ, (ii) the anyon type $A = (C, \chi)$, where C is the conjugacy class and χ is an irreducible representation, (iii) the local internal state $u = (v, i)$ at the start, and (iv) the local internal state $u' = (v', j)$ at the end, where u, v are elements in the conjugacy class C and i, j are the matrix indices for the irreducible representation χ. In general, a ribbon operator creates vertex and plaquette violations at both ends of the ribbon.

The easiest way to express the ribbon operator $F_\rho^{A;u,u'}$ is in terms of basis elements $F_\rho^{(z,v)}$, whose action is depicted in Fig. 1b.

$$F^{A;u,u'} = \frac{|\chi|}{|\mathbf{Z}(r)|} \sum_{n \in \mathbf{Z}(r)} \Gamma^\chi_{jj'}(n) F^{(q_v n \bar{q}_{v'}, v)}, \quad A = (C, \chi),\ u = (v, j) \tag{S12}$$

Here, $r \in C$ is a representative element of the conjugacy class corresponding to the anyon in question. $\mathbf{Z}(r)$ is the centralizer of r, the subgroup of all elements in that commute with r. Furthermore, we have

$$v = q_v r \bar{q}_v, \quad v' = q_{v'} r \bar{q}_{v'} \tag{S13}$$

The operator $F_\rho^{A;u,u'}$ creates two excitations at either end of ρ, but commutes with all stabilizers along the bulk of the ribbon. The anyons on either end will be of type $A = (C, \chi)$, and the internal state of each anyon will be labeled by u and u', respectively.

Ribbon operators are constructed out of fundamental building blocks known as "triangle" operators: these are operators that act on a single edge of the lattice, and are strung together through a recursive formula to yield a full-length ribbon operator. Fig. 2 contains a table listing the eight different triangle operators, distinguished by three different characteristics: (i) whether they act on the direct or dual lattice, (ii) whether the direction of the operator is aligned with the lattice, and (iii) a property known as "local orientation" [S6].

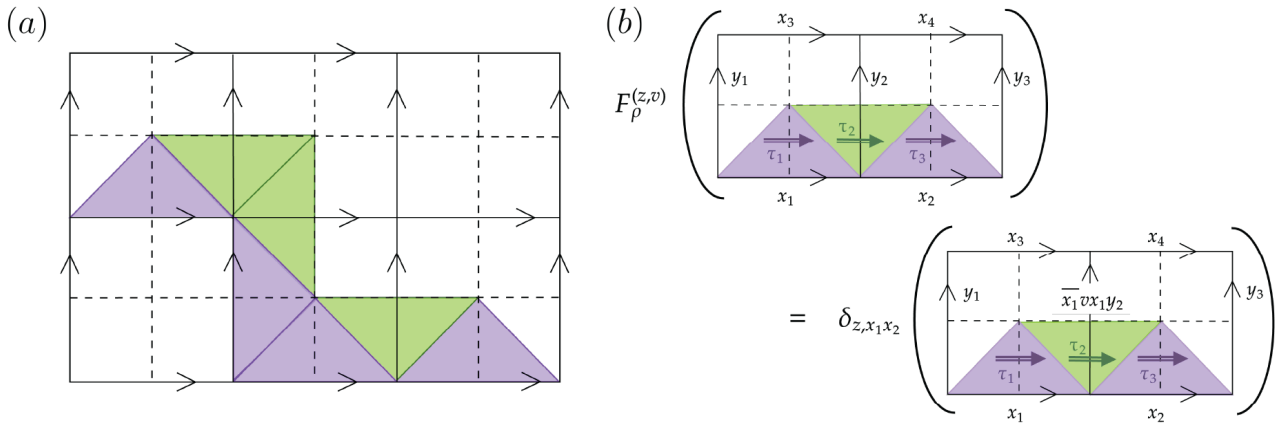


Figure 1. (a) Ribbon operators create non-Abelian anyons at their ends; they are thicker as compared to their abelian counterparts, and act both on the direct and dual edges of the lattice. (b) The action of a ribbon operator in the "group element" basis. This ribbon creates an anyon with local flux v and nonlocal flux $zv\bar{z}$, and will generically create a superposition of charges. The anyon basis ribbons are superpositions of these ribbons.

A full-length ribbon operator $F^{(z,v)}$ is built out of triangle operators recursively. Consider the three triangle ribbon pictured in Fig. 1. Denote the triangles by τ_i. We can derive the action of the full ribbon using the following formula:

$$F^{(z,v)}(\rho = \rho_1 \cup \rho_2) = \sum_{k\in G} F^{(k,v)}(\rho_1) F^{(\overline{k}z,\overline{k}vk)}(\rho_2)\ . \tag{S14}$$

Triangles τ_1 and τ_3 act on the direct lattice, are aligned with the lattice orientation, and have counter-clockwise local orientation, meaning the corresponding triangle operators $F^{(z,v)}(\tau_1)$ and $F^{(z,v)}(\tau_3)$ act by $|x\rangle \to \delta_{z,x}|x\rangle$. Triangle τ_2 acts on the dual lattice, is opposite to the lattice orientation, and has counter-clockwise local orientation, and so the corresponding operator acts as $F^{(z,v)}(\tau_2)|x\rangle = \delta_{z,e}|vx\rangle$.

We start by breaking up the ribbon into two pieces: $\rho = \tau_1 \cup (\tau_2 \cup \tau_3)$:

$$F^{(z,v)}(\rho) = \sum_{k\in G} F^{(k,v)}(\tau_1) F^{(\overline{k}z,\overline{k}vk)}(\tau_2 \cup \tau_3) = \sum_{k\in G} \delta_{x_1,k} F^{(\overline{k}z,\overline{k}vk)}(\tau_2 \cup \tau_3) = F^{(\overline{x}_1 z,\overline{x}_1 v x_1)}(\tau_2 \cup \tau_3). \tag{S15}$$

Applying the recursion relation again:

$$\begin{aligned} F^{(z,v)}(\rho) &= \sum_{k\in G} F^{(k,\overline{x}_1 v x_1)}(\tau_2) F^{(\overline{k}\overline{x}_1 z,\overline{k}\overline{x_1} v x_1 k)}(\tau_3) \\ &= \sum_{k\in G} \delta_{k,e} |(\overline{x}_1 v x_1) y_2\rangle \langle y_2| \, \delta_{x_2,\overline{k}\overline{x}_1 z} \\ &= \delta_{x_1 x_2, z} |(\overline{x}_1 v x_1) y_2\rangle \langle y_2|\ . \end{aligned} \tag{S16}$$

This is the same action quoted in Fig. 1(b).

2. *Unitary ribbon*

We now specialize to useful ribbons used in the main text. We observe that by summing over the local internal state on one end of the ribbon, the resulting ribbon operator is unitary. Such unitary ribbon can be decomposed into three steps: 1) ungauging, 2) finite-depth local unitary (FDLU) circuit in a simpler TO, and 3) regauging [S7]. For the ungauging step, it is a sequential unitary circuit that "unzips" along the length of the ribbon to a simpler TO. Then, the FDLU circuit is either a string operator or symmetry fractionalization pattern in the simpler TO. Finally, the regauging circuit "zips up" along the length of the ribbon back to the original TO.

The unitary ribbon for a [2] charge vacuum pair is given by:

$$F^{(+,[2]);0} = \sum_j F_\rho^{(+,[2]);0,j} = \prod_{x\in\rho} \mathcal{Z}_x^{\prod_{g<x} Z_g}. \tag{S17}$$

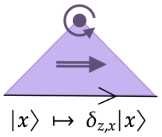
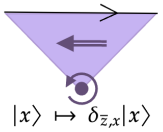
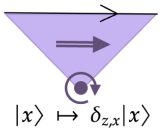
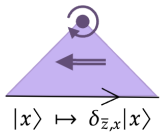
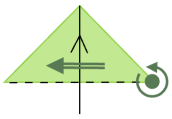
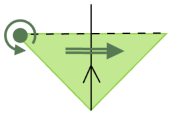
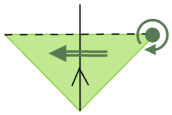
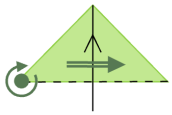
		Alignment	
Type of Triangle	Local Orientation	Aligned	Opposite
Direct	Counterclockwise	 $\|x\rangle \mapsto \delta_{z,x}\|x\rangle$	 $\|x\rangle \mapsto \delta_{\bar{z},x}\|x\rangle$
	Clockwise	 $\|x\rangle \mapsto \delta_{z,x}\|x\rangle$	 $\|x\rangle \mapsto \delta_{\bar{z},x}\|x\rangle$
Dual	Counterclockwise	 $\|x\rangle \mapsto \delta_{z,e}\|x\bar{v}\rangle$	 $\|x\rangle \mapsto \delta_{z,e}\|vx\rangle$
	Clockwise	 $\|x\rangle \mapsto \delta_{z,e}\|\bar{v}x\rangle$	 $\|x\rangle \mapsto \delta_{z,e}\|xv\rangle$

Figure 2. Definition of triangle operators, based on the type (direct or dual), alignment (aligned or opposite), and local orientation (clockwise or counterclockwise) of the triangle operator [S2]. Direct and dual edge is denoted by solid and dotted line respectively. The orientation of the edge is indicated by the single arrow; the direction of the ribbon is indicated by the double arrow. The local orientation of the triangle operator is determined by first fixing the hinge (dark circle) at the plaquette center touching the triangle operator, and then turn along the ribbon direction.

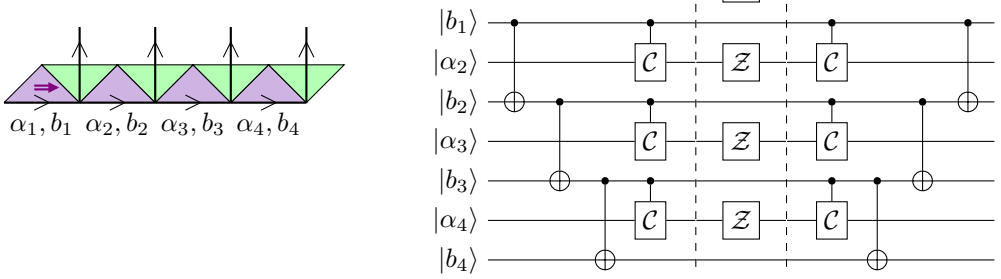
Here x, g are edges on the lattice along the ribbon ρ, and $g < x$ means that g occurs before edge x, traveling along ρ from the start of the ribbon to the end. The gate $\mathcal{Z}_x^{Z_g}$ is a controlled gate with the qutrit on edge x as the target and the qubit on edge g as the control:

$$\mathcal{Z}_x^{Z_g} \left|\mu^a\right\rangle_x \left|\sigma^b\right\rangle = \omega^{(-1)^b a} \left|\mu^a\right\rangle_x \left|\sigma^b\right\rangle, \quad a = 0, 1, 2, \quad b = 0, 1. \tag{S18}$$

There can also be multiple controls on some set of edges $\{g\}$:

$$\mathcal{Z}_x^{\prod_g Z_g} \left|\mu^a\right\rangle_x \prod_g \left|\sigma^{b_g}\right\rangle_g = \omega^{(-1)^{\sum_g b_g} a} \left|\mu^a\right\rangle_x \prod_g \left|\sigma^b\right\rangle_g, \quad a = 0, 1, 2, \quad b_g = 0, 1. \tag{S19}$$

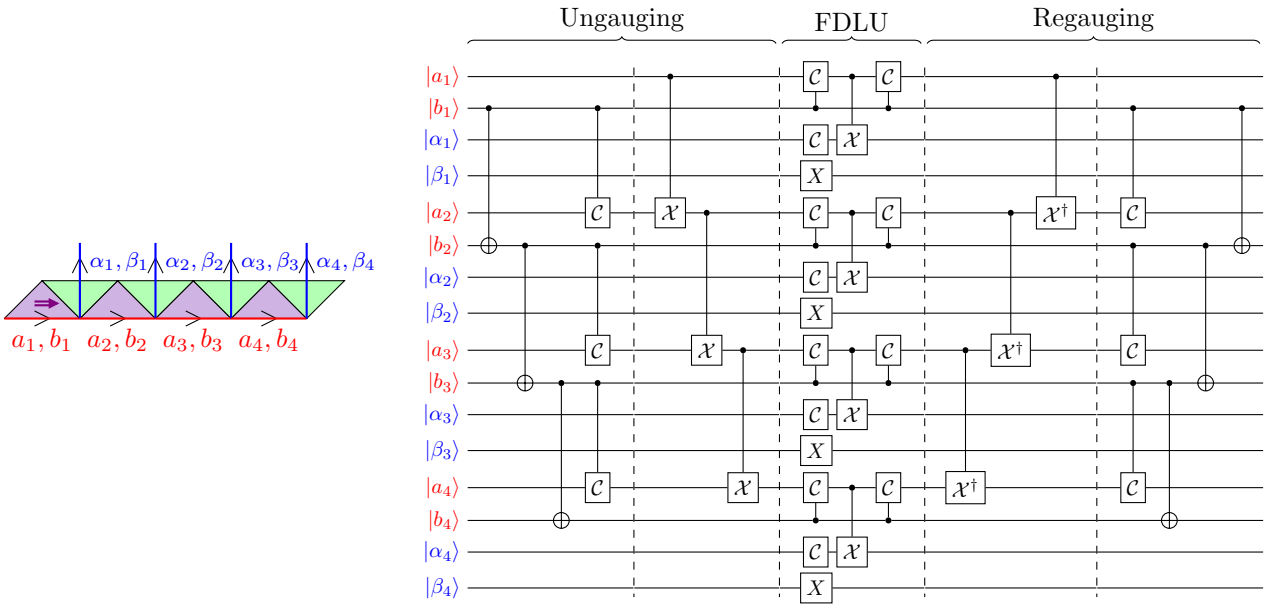
$F^{(+,[2]);0}$ can be expressed as a circuit by using the identities in Section S1 B (where the purple arrow in the first purple triangle indicates the direction of the ribbon):



(S20)

The [2] charge ribbon operator circuit above illustrate how a unitary ribbon operator can be decomposed into three steps (as divided by dotted line): 1) ungauging, 2) finite-depth local unitary (FDLU) circuit in a simpler TO, and 3) regauging [S7]. For the ungauging step, it is a sequential unitary circuit that "unzip" the direct edges to a simpler TO (in this case ungauging to $\mathbb{Z}_3$ toric code). Then, the FDLU circuit is just the usual string operator for the e anyons in $\mathbb{Z}_3$ toric code. Finally, the regauging layer is used to "zip up" the direct edges back to S_3 TO.

Unitary ribbons for a $[C_2]$ flux is more complicated that the [2] charge ribbon. For example, to implement the unitary ribbon $F^{C_2;\sigma} = \sum_{c\in C_2} F^{C_2;\sigma,c}$, where the local internal state is $|\sigma\rangle$ at the start of the ribbon, the circuit for a C_2 ribbon (with direction indicated by the purple arrow in the first purple triangle) with counterclockwise local orientation (the issue of local orientation will be explained below) is given by:

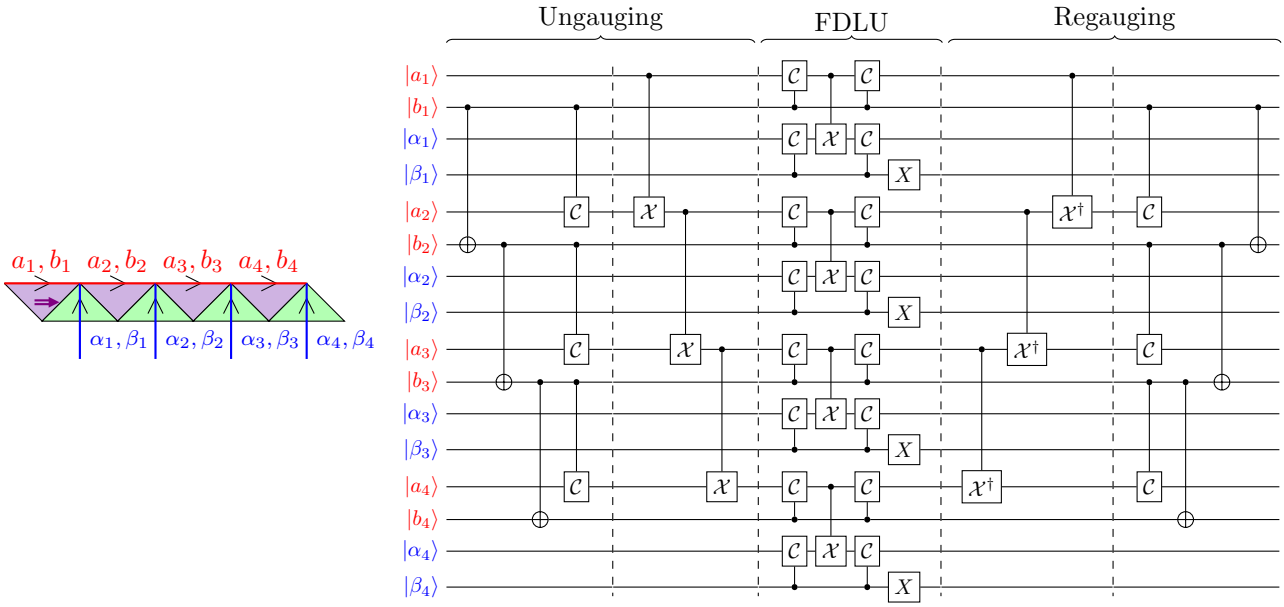


(S21)

Note that the ungauging and regauging steps are more complicated than that of the [2] charge ribbon, because it does not stop at the level of the $\mathbb{Z}_3$ toric code but continue to the level of $\mathbb{Z}_3$ paramagnet [S7].

As a remark, to implement a C_2 ribbon where the starting local internal state is $|\mu\sigma\rangle$ and $|\bar{\mu}\sigma\rangle$, the entire circuit needed to be conjugated with $\mathcal{X}$ and $\mathcal{X}^\dagger$ respectively on the first direct edge (a_1, b_1 in this case), as it modifies the action of individual dual triangle to $L^{\mu\sigma}$ and $L^{\bar{\mu}\sigma}$ respectively (for the counterclockwise, aligned case).

It is important to keep track of the local orientation of the ribbon to ensure consistent action of ribbon operator [S2, S6]. The example ribbons displayed above are of counterclockwise local orientation. The circuit for a C_2 ribbon with clockwise local orientation is given by:



(S22)

The circuit implementation for clockwise local orientation can be derived from that of counterclockwise local orientation by applying the inverse operation circuit (Eq. S5) on the dual edges.

3. *Generalized ribbon operator*

In Ref. S2, a generalized ribbon operator is introduced that relaxes the condition that a ribbon operator only create violations at two sites. It is observed the charge violations can be decoupled from the flux violations at the ends of a ribbon; i.e., the charge violations can be moved arbitrarily far away from the flux violations. This can be understood from the gauging perspective discussed in Section S1 D 2: the ungauging/regauging circuit can be along a length beyond the support of the FDLU part of the circuit. Then the charge violations are located at the starting and ending location of the ungauging/regauging circuit, whereas the flux violations are located at the starting and ending location of the FDLU circuit.

Generalized ribbon operators are crucial for defining the absolute logical encoding used in demonstrating the universal gate set in Section IV B, IV C, and IV D. A generalized ribbon operator ensures global fluxes are well-defined (so that they can be compared consistently) and allows the fluxes at the two ends of a ribbon operator to be at arbitrary locations and moved for braiding. When defining a flux, we need to specify the base point where the flux loop starts and ends. Changing base point conjugate the flux by the value of the path between the old and new base point. Therefore, to compare the flux internal states between distinct fluxes, the same base point has to be used to ensure consistency.

For the experiment, we fix the base point to be at the origin such that all global fluxes are defined with respect to the origin; an example is shown in Fig. 1(k).

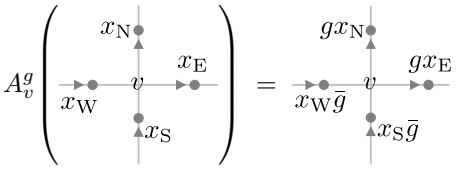
S2. QUBIT–QUTRIT ENCODING OF $\mathcal{D}(S_3)$ STABILIZERS AND PROJECTORS

In Section S1 B we detailed the encoding of S_3 group elements in terms of qubit–qutrit pairings and described the corresponding group multiplication rules. Here we provide a description of the *standard* $\mathcal{D}(S_3)$ projectors, originally defined in Ref. S3, expressed in terms of their action on the qubit and qutrit degrees of freedom. In particular, we derive explicit expressions for the $\mathcal{D}(S_3)$ vertex and plaquette projectors in terms of the operators $A^{\mathbb{Z}_2}$, $A^{\mathbb{Z}_3}$, $B^{\mathbb{Z}_2}$, and $B^{\mathbb{Z}_3}$ introduced in Eqs. (1) and (2). Similar definitions for the W-flux and non-contractible projectors are also given.

We begin by writing $\mathcal{D}(S_3)$ hamiltonian

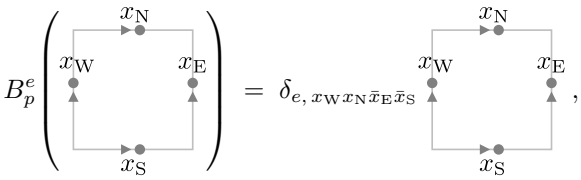
$$H = -\sum_v A_v - \sum_p B_p^e, \tag{S23}$$

where the vertex projector $A_v = \frac{1}{6}\sum_{g\in S_3} A_v^g$ enforces the zero-charge constraint at each vertex. Pictorially, the action of A_v^g on a group-element configuration is



(S24)

Here, the solid circle and triangle on each edge denote the qubit–qutrit encoding. Similarly, the plaquette projectors B_p^e enforce trivial total flux around each plaquette. Their action on basis states is

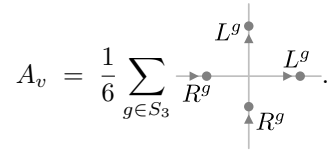


(S25)

where $x_{\mathrm{W}}x_{\mathrm{N}}\bar{x}_{\mathrm{E}}\bar{x}_{\mathrm{S}}$ denotes the flux around the plaquette.

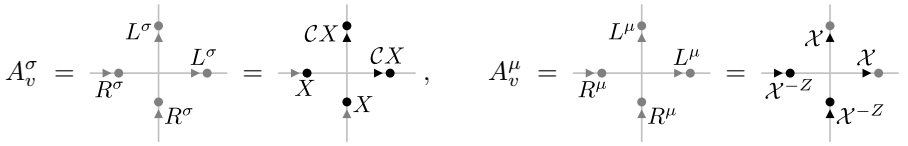
A. Vertex projectors

From Eq. (S24), A_v^g left-multiplies the group elements on the north (N) and east (E) edges, and right-multiplies those on the south (S) and west (W) edges. The full vertex projector can be written as



(S26)

Using Eqs. (S2) and (S3), the terms A_v^σ and A_v^μ can be expressed as



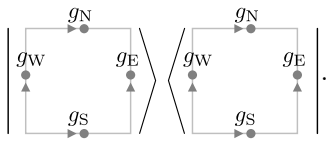
(S27)

Expanding A_v, we obtain

$$
\begin{aligned}
A_v &= \frac{1}{6}\left(1 + A^\mu + A^{\bar{\mu}} + A^\sigma + A^{\mu\sigma} + A^{\bar{\mu}\sigma}\right) \\
&= \frac{1}{6}\left(1 + A^\sigma\right)\left(1 + A^\mu + A^{\bar{\mu}}\right) \\
&= \frac{1}{2}\left(1 + \;\; \mathcal{C}X \;\; \mathcal{C}X \;\; X \;\; X\right) \cdot \frac{1}{3}\left[1 + \left(\;\; \mathcal{X} \;\; \mathcal{X} \;\; \mathcal{X}^{-Z} \;\; \mathcal{X}^{-Z} \;\; + \text{h.c.}\right)\right] \\
&= A^{\mathbb{Z}_2}\, A^{\mathbb{Z}_3}.
\end{aligned}
\tag{S28}
$$

B. Plaquette projector

Since B_p^e enforces trivial flux around each plaquette, it can be written as

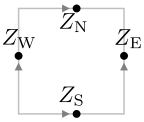


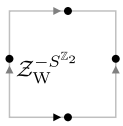
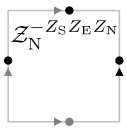
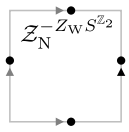
(S29)

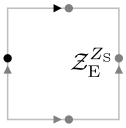
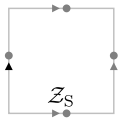
Using the qubit–qutrit encoding $g = \mu^a\sigma^b = \sigma^b\mu^{(-1)^b a}$ and $\bar{g} = \sigma^b\mu^{-a} = \mu^{-(-1)^b a}\sigma^b$, the flux $g_{\rm S}g_{\rm E}\bar{g}_{\rm N}\bar{g}_{\rm W}$ becomes

$$\begin{aligned}\text{flux} &= \mu^{a_{\rm S}}\sigma^{b_{\rm S}}\ \mu^{a_{\rm E}}\sigma^{b_{\rm E}}\ \sigma^{b_{\rm N}}\mu^{-a_{\rm N}}\ \sigma^{b_{\rm W}}\mu^{-a_{\rm W}}\\ &= \mu^{a_{\rm S}}\,\mu^{(-1)^{b_{\rm S}}a_{\rm E}}\,\sigma^{b_{\rm S}+b_{\rm E}+b_{\rm N}}\,\mu^{-a_{\rm N}}\,\mu^{-(-1)^{b_{\rm W}}a_{\rm W}}\,\sigma^{b_{\rm W}}\\ &= \mu^{a_{\rm S}}\,\mu^{(-1)^{b_{\rm S}}a_{\rm E}}\,\mu^{-(-1)^{b_{\rm S}+b_{\rm E}+b_{\rm N}}a_{\rm N}}\,\mu^{-(-1)^{b_{\rm S}+b_{\rm E}+b_{\rm N}+b_{\rm W}}a_{\rm W}}\,\sigma^{b_{\rm S}+b_{\rm E}+b_{\rm N}+b_{\rm W}}.\end{aligned} \tag{S30}$$

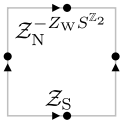
Each term in the product can be written in terms of Z and $\mathcal{Z}$ operators on the qubit and qutrit degrees of freedom, that is,

$\sigma^{b_{\rm S}+b_{\rm E}+b_{\rm N}+b_{\rm W}} \rightarrow$  $:= S^{\mathbb{Z}_2}$, (S31)

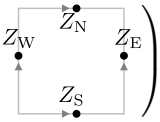
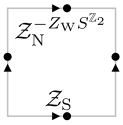
$\mu^{-(-1)^{b_{\rm S}+b_{\rm E}+b_{\rm N}+b_{\rm W}}a_{\rm W}} \rightarrow$  , $\mu^{-(-1)^{b_{\rm S}+b_{\rm E}+b_{\rm N}}a_{\rm N}} \rightarrow$  $=$  ,

$\mu^{(-1)^{b_{\rm S}}a_{\rm E}} \rightarrow$  , and $\mu^{a_{\rm S}} \rightarrow$  . (S32)

Collecting all contributions, the flux around the plaquette can be measured by

$S^{\mathbb{Z}_2} \cdot \mathcal{Z}_{\rm w}^{-S^{\mathbb{Z}_2}}$  $\mathcal{Z}_{\rm E}^{Z_{\rm S}} = S^{\mathbb{Z}_2} \cdot S^{\mathbb{Z}_3}$. (S33)

Moreover, projectors onto the $+1$ eigenspaces of $S^{\mathbb{Z}_2}$ and $S^{\mathbb{Z}_3}$ enforce the trivial flux constraint, and Eq. (S29) can be written as (see Ref. S8):

$$\begin{aligned}B_p^e &= \frac{1}{2}\left(1 + \right.\ \text{  }\ \left.\right) \cdot \frac{1}{3}\left[1 + \left(\ \text{  }\ + \text{h.c.}\right)\right]\\ &= B^{\mathbb{Z}_2}\,B^{\mathbb{Z}_3}.\end{aligned} \tag{S34}$$

C. W-flux and non-contractible Z projectors of $\mathcal{D}(S_3)$

The W-flux is the topological flux defined by the ordered group product along a closed loop, based at a fixed origin, that encircles an excitation. Fig. 3 enumerates the different paths used to define the W-flux stabilizer S_{W_p}, each of which encloses exactly one plaquette p. The corresponding projector W_p is defined as the projector onto the $+1$ eigenspace of S_{W_p}. Using steps analogous to those employed in deriving the plaquette operator B_p, the W-flux projector can also be written as

$$W_p = W_p^{\mathbb{Z}_2}\,W_p^{\mathbb{Z}_3}. \tag{S35}$$

Here $W_p^{\mathbb{Z}_2} = B_p^{\mathbb{Z}_2}$ (as defined in Eq. (S31)) for the paths listed in Fig. 3, while $W_p^{\mathbb{Z}_3}$ has a form analogous to $S_B^{\mathbb{Z}_3}$ and acts non-trivially on all qudits along the loop.

Correlators of the W-flux are obtained by forming products of stabilizers, for example $S_{W_{p_1}}S_{W_{p_2}}$, and projecting onto their joint $+1$ eigenspace, yielding the projector $\Pi_{W_{p_1}W_{p_2}}$. This projector similarly factorizes into independent $\mathbb{Z}_2$ and $\mathbb{Z}_3$ components.

$$\Pi_{W_{p_1}W_{p_2}} = \Pi^{\mathbb{Z}_2}_{W_{p_1}W_{p_2}}\Pi^{\mathbb{Z}_3}_{W_{p_1}W_{p_2}} \tag{S36}$$

Similarly, the 'Z-logical' projectors of the $\mathcal{D}(S_3)$ code are defined as the ordered product of group elements along non-contractible loops. For the horizontal and vertical directions, these take the form $Z_{\rm hori} = Z^{\mathbb{Z}_2}_{\rm hori}Z^{\mathbb{Z}_3}_{\rm hori}$ and $Z_{\rm vert} = Z^{\mathbb{Z}_2}_{\rm vert}Z^{\mathbb{Z}_3}_{\rm vert}$, respectively (see Fig. 1e–f for the corresponding stabilizer definitions).

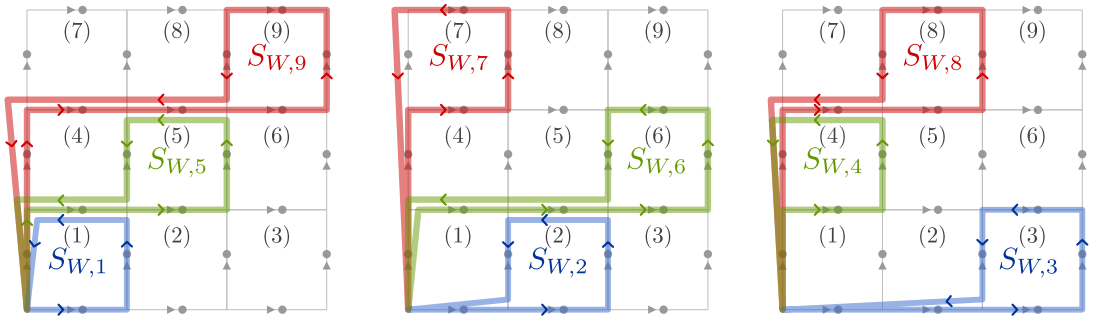


Figure 3. **W-flux.** Paths along which W-flux stabilizers are evaluated to benchmark the experimental results presented in the main text.

S3. COHERENT MOVING OF NON-ABELIAN ANYONS: THEORY

In our experiment, we only need to coherently move non-Abelian pure fluxes. We will focus on the protocol for coherently moving non-Abelian flux for the rest of the section. This is a special case of the general procedure of coherent moving [S9].

Suppose we want to coherently move an anyon from location A to B. Operationally, we demand a deterministic process such that there is no remnant anyon at the site A after coherent moving. We also require that, at no point in the protocol, is the global flux label (i.e. what defines the "absolute" logical encoding) stored locally; the coherent moving protocol only involves the local flux label instead. We implement coherent moving unitarily with the use of one ancilla qudit, where d matches the quantum dimension of the non-Abelian anyon.

Ribbon operators create pairs of excitation at their end points; to ensure that an anyon is deterministically moved from the starting site to the ending site, we need to ensure that the fusion outcome is always vacuum at the starting site. This is done by judicious choice of internal state at the ends of a ribbon operator. For a pair of non-Abelian fluxes, the state in the vacuum fusion channel is the symmetric superposition over all representatives in a conjugacy class C, i.e.,

$$|\text{vac}\rangle = \frac{1}{|C|}\sum_{c\in\mathcal{C}} |c,\bar{c}\rangle\,. \tag{S37}$$

The pairing of $c,\bar{c}$ ensures flux neutrality since the total flux is $c\cdot\bar{c}=e$; the symmetrization ensures charge neutrality as any permutation of the terms leave the state invariant, so the state is not charged under any symmetry action (which acts by conjugation). For moving non-Abelian *pure* flux, the procedure is simplified because pure fluxes already satisfy the property that the local internal flux state is a symmetric superposition over all possible basis states. Therefore, we only need to ensure the correct pairing $c,\bar{c}$ for the internal state of the ribbon operators.

The coherent moving protocol of non-Abelian flux consists of three main steps:

1. entangle the ancilla with the local internal state of the flux;

2. conditioned on the ancilla, apply the corresponding ribbon operator;

3. disentangle ancilla.

For step 1, a $C_L\mathcal{X}_a$ gate is implemented to entangle the ancilla with the local internal state of the flux.

For step 2, a C_2 ribbon operator is conjugated by $C_a\mathcal{X}_{dir}$, where *dir* is the first direct edge of the ribbon operator, such that starting local internal state created by the ribbon operator matches with the ancilla state. This ensures that the ribbon operator create the same C_2 flux at the starting site to cancel out the flux there.

For step 3, we want to implement a $C_L\mathcal{X}_a$ gate again to disentangling the ancilla from the local internal state of the flux at the start. However, the flux at the start is annihilated to vacuum! The key is to recognize that the flux label *based at the same vertex* remains the same due to the fact that ribbon operator can only create vacuum pair of anyons, i.e. ribbon operator satisfies the neutrality condition by construction [S2]. Therefore, the control qutrit is the global internal state (based at the starting vertex of the ribbon operator) of the new C_2 flux at the ending site.

See Section S10 A and Section S10 B for examples of coherent moving circuit implementations.

S4. PHYSICAL ENCODING AND IMPLEMENTATION OF PRIMITIVE GATES FOR $\mathcal{D}(S_3)$

This Section describes our implementation of qudit encoding and gates, realized with the native gate set of Quantinuum H2-series devices [S10, S11]. The native gates for the H2 architecture are as follows:

$$U_{1q}(\alpha,\beta) = e^{-\frac{1}{2}i\pi\beta Z}\; e^{-\frac{1}{2}i\pi\alpha X}\; e^{+\frac{1}{2}i\pi\beta Z}, \quad R_Z(\alpha) = e^{-\frac{1}{2}i\pi\theta Z} \quad \text{and} \tag{S38}$$

$$\text{ZZPhase}(\theta) = e^{-\frac{1}{2}i\pi\theta(Z\otimes Z)}. \tag{S39}$$

The number of ZZPhase gates significantly affects the overall implementation cost and serves as a key metric for evaluating gate decompositions.

Each six-dimensional qudit, representing the $\mathcal{D}(S_3)$ degrees of freedom on the edges of our square lattice, is encoded using a qutrit and a qubit. The qutrit, in turn, is encoded using two qubits. Therefore, each qudit requires a total of 3 qubits. The qutrit encoding is defined as:

$$|0\rangle_{\blacktriangle} := |00\rangle\,, \quad |1\rangle_{\blacktriangle} := |10\rangle\,, \quad |2\rangle_{\blacktriangle} := |11\rangle\,, \quad |\text{nc}\rangle := |01\rangle. \tag{S40}$$

The non-computational state, $|\text{nc}\rangle$, is used for error detection; its presence indicates that the two encoding qubits have fallen outside the qutrit subspace.

Qutrits are primarily manipulated using the $\mathcal{X}$, $\mathcal{Z}$, and $\mathcal{C}$ gates, as defined in Eqs. (S4). $\mathcal{X}$ and $\mathcal{Z}$ satisfy the commutation relation $\mathcal{X}\mathcal{Z} = \omega\mathcal{Z}\mathcal{X}$. The charge-conjugation gate $\mathcal{C}$ is Hermitian, $\mathcal{C}^\dagger = \mathcal{C}$, and conjugates $\mathcal{X}$ and $\mathcal{Z}$:

$$\mathcal{C}\mathcal{X}\mathcal{C} = \mathcal{X}^\dagger \quad \text{and} \quad \mathcal{C}\mathcal{Z}\mathcal{C} = \mathcal{Z}^\dagger. \tag{S41}$$

The qutrit $\mathcal{Z}$ gate can be efficiently implemented using two single-qubit native R_Z gates:

$$\mathcal{Z} := \quad = \bar{\omega}\ \text{diag}(1,\ \omega,\ \omega,\ \bar{\omega}). \tag{S42}$$

The action of $\mathcal{Z}$ on $|\text{nc}\rangle$ is trivial (up to ω phase). The extra $\bar{\omega}$ phase needs to be corrected to properly implement a controlled-$\mathcal{Z}$ (C$\mathcal{Z}$) gate. In contrast, implementing $\mathcal{X}$ requires either two CNOT gates or a single ZZPhase gate:

$$\mathcal{X} := \quad = \quad = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{S43}$$

The charge-conjugation gate $\mathcal{C}$ can also be implemented using only a single ZZPhase gate.

$$\mathcal{C} := \quad = \quad = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{S44}$$

$\mathcal{X}$ can be transformed into $\mathcal{Z}$ gate by performing a qutrit Fourier transform, denoted by $\mathcal{H}$ which acts on the qutrit space as follows:

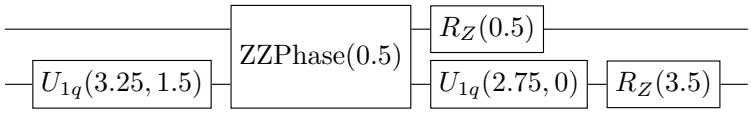
$$\mathcal{H}\,|i\rangle_{\blacktriangle} = \frac{1}{\sqrt{3}}(|0\rangle_{\blacktriangle} + \omega^i\,|1\rangle_{\blacktriangle} + \omega^{2i}\,|2\rangle_{\blacktriangle}), \tag{S45}$$

The native implementation of $\mathcal{H}$ requires three ZZPhase gates, preserving the $|\text{nc}\rangle$ state. Note that we treat the action of $\mathcal{H}$ on $|0\rangle_{\blacktriangle}$ as a *state preparation procedure*, which can be implemented using one ZZPhase gate:

$$\mathcal{H}\,|00\rangle = \quad = \frac{1}{\sqrt{3}}(|00\rangle + |10\rangle + |11\rangle)\,. \tag{S46}$$

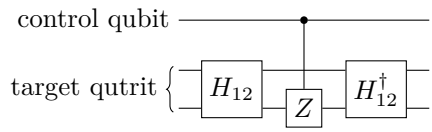
Furthermore, $\mathcal{H}$ commutes with $\mathcal{C}$, since $\mathcal{H}^2 = \mathcal{C}$.

Similarly, H_{12} diagonalizes $\mathcal{C}$, leaving $|0\rangle_{\blacktriangle}$ unchanged while applying a "Hadamard gate" to both $|1\rangle_{\blacktriangle}$ and $|2\rangle_{\blacktriangle}$. In terms of native gates, H_{12} requires a single ZZPhase gate:

$H_{12} :=$ 

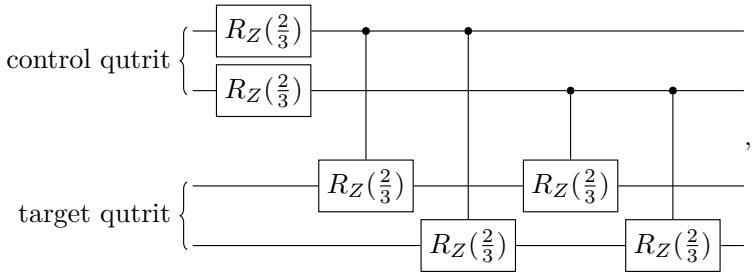
$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}.$ (S47)

Since $\mathcal{C}$ squares to the identity, the C$\mathcal{C}$ gate is defined with a qubit control and qutrit target. To avoid applying three qubit CCZ gate to implement C$\mathcal{C}$, we use the fact that $H_{\mathcal{C}}$ disentangles the $\mathcal{C}$ gate, i.e., $H_{12}\ \mathcal{C}\ H_{12}^{\dagger} = IZ$. The implementation of C$\mathcal{C}$ requires three ZZPhase gates:

C$\mathcal{C}$:= 

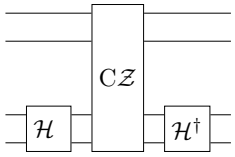
(S48)

The application of C$\mathcal{Z}$ requires four CR_z gates:

C$\mathcal{Z}$:= 

, (S49)

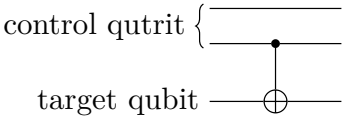
where each of the CR_z gate can be implemented using a single ZZPhase gate. The initial single qubit R_z gates on control qutrit compensate for the extra phase $\bar{\omega}$ in Eq. (S43). Applying C$\mathcal{Z}$ while transforming the target qutrit by $\mathcal{H}$ yields the C$\mathcal{X}$ gate:

C$\mathcal{X}$:= 

(S50)

The C$\mathcal{X}$ gate requires a total of $3 + 4 + 3 = 10$ ZZPhase gates.

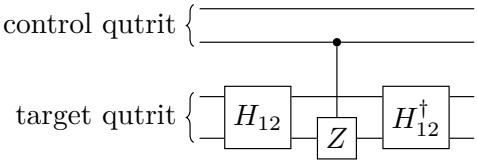
We define a C_2NOT gate with a qutrit as the control and a qubit as the target.

C_2NOT := 

(S51)

It acts trivially on the target qubit when the control qutrit is in the $|0\rangle$ or $|1\rangle$ state, and flips the target qubit when the control is in the $|2\rangle$ state. It is implemented by a single CNOT gate conditioned on the second physical qubit of the control qutrit, because in the physical encoding of the qutrit (see Eq. (S40)), the second physical qubit is in the $|1\rangle$ state only for the $|2\rangle = |11\rangle$ qutrit state.

Moreover, we define the CX_{12} gate with a qutrit control and a qutrit target.

CX_{12} := 

(S52)

The gate acts as the identity when the control qutrit is in the $|0\rangle$ or $|1\rangle$ state. When the control is in the $|2\rangle$ state, it swaps the target states $|1\rangle$ and $|2\rangle$ while leaving $|0\rangle$ invariant.

S5. CIRCUIT OPTIMIZATION BY LOCAL BASIS TRANSFORMATION

For each experiment, before compiling the circuit unitary U into native gates using TKET [S12], we locally transform every qutrit to the dual ($\mathcal{X}$) basis by applying $\mathcal{H}$, as defined in Eq. (S45), and construct $\tilde{U}$ such that

$$\tilde{U} = \mathcal{H}^{\otimes n\dagger}\, U\, \mathcal{H}^{\otimes n}, \tag{S53}$$

where n is the number of qutrits. We then compile and execute $\mathcal{H}^{\otimes n}\,\tilde{U}\,\mathcal{H}^{\otimes n}$ instead of compiling and executing U directly. This local basis transformation reduces the number of gates after compilation and, empirically, also reduces memory errors. For comparison,

$$\begin{aligned}
\mathcal{X}\text{-basis measurement:}\quad & \begin{matrix} n_{2q}\text{ gates} = 792 \\ \text{depth}_{2q} = 261 \end{matrix} \xrightarrow[\text{transformation}]{\text{local basis}} \begin{matrix} 744 \\ 232 \end{matrix}, \text{ and} \\
\text{Pull-through gate:}\quad & \begin{matrix} n_{2q}\text{ gates} = 911 \\ \text{depth}_{2q} = 354 \end{matrix} \xrightarrow[\text{transformation}]{\text{local basis}} \begin{matrix} 845 \\ 307 \end{matrix}.
\end{aligned} \tag{S54}$$

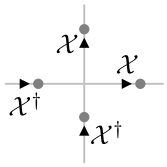
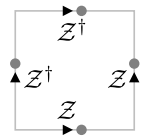
It is straightforward to compute $\tilde{U}$ because $\mathcal{H}$ commutes with the charge-conjugation gate $\mathcal{C}$ and maps a C$\mathcal{X}$ gate to another C$\mathcal{X}$ gate with the control and target qutrits swapped.

S6. GROUND STATE PREPARATION

This section details the methodology for preparing the ground state of the $\mathcal{D}(S_3)$ model. The preparation strategy we use follows the method given in the appendix of Ref. S13 and consists of a multi-step procedure. The first step is the preparation of the $\mathbb{Z}_3$ toric code ground state. Subsequently, this Abelian state is transformed into the desired non-Abelian $\mathcal{D}(S_3)$ order by gauging its charge-conjugation symmetry. A detailed exposition of these steps follows.

A. Preparation of $\mathbb{Z}_3$ toric code ground state

The ground state of the $\mathbb{Z}_3$ toric code with qutrits on the edges of the square lattice is stabilized by,

$S_{\mathcal{A}} =$  , $S_{\mathcal{B}} =$  . (S55)

While these stabilizers can be prepared in constant depth using an adaptive circuit with mid-circuit measurements and ancilla qutrits, this approach incurs a significant qubit overhead. On the Quantinuum System Model H2, which has 56 qubits, preparing the $\mathcal{D}(S_3)$ ground state on a 3×3 lattice already requires 54 qubits, leaving little room for additional ancillae. We therefore employ a purely unitary preparation protocol that requires neither mid-circuit measurements nor ancillae for the $\mathbb{Z}_3$ toric code state preparation itself. The protocol is described below.

1. All qutrits are initialized in the state $|0\rangle$, which trivially satisfies all plaquette stabilizers $S_{\mathcal{B}} = +1$.

2. To satisfy the vertex stabilizers $S_{\mathcal{A}} = +1$, we proceed in an iterative manner:

   (i) Select a vertex stabilizer and designate one of its qutrits as the"representative." The representative qutrit is prepared in the state $|+\rangle_{\blacktriangle} := \frac{1}{\sqrt{3}}(|0\rangle_{\blacktriangle} + |1\rangle_{\blacktriangle} + |2\rangle_{\blacktriangle})$, using the state-preparation circuit of Eq. (S46).

   (ii) Apply a sequence of C$\mathcal{X}$ or C$\mathcal{X}^\dagger$ gates from the representative qutrit to the remaining qutrits within the same vertex stabilizer. A C$\mathcal{X}$ gate is applied when both the control and target qutrits are acted on by $\mathcal{X}\mathcal{X}$ or $\mathcal{X}^\dagger\mathcal{X}^\dagger$ in the vertex stabilizer; otherwise, a C$\mathcal{X}^\dagger$ gate is applied.

   This procedure is repeated for all vertex stabilizers except one, ensuring that a qutrit is selected as a representative only if it is in the $|0\rangle$ state and has not been acted upon previously. The final vertex stabilizer is implicitly satisfied due to the global constraint $\prod_v S_{\mathcal{A}_v} = 1$. The vertex preparation order used to obtain the $\mathbb{Z}_3$ toric code ground state on 3×2 and 3×3 lattices is illustrated in Fig. 4a.

The resulting state is the logical ground state $|00\rangle_L$ of the $\mathbb{Z}_3$ toric code. To prepare the logical state $|{+}{+}\rangle_L$, where both logical $\mathcal{X}_L$ operators have eigenvalue $+1$, a dual procedure is employed. All qutrits are first initialized in the $|+\rangle_{\blacktriangle}$ state, satisfying all vertex stabilizers $S_{\mathcal{A}} = +1$. Then, plaquette stabilizers $S_{\mathcal{B}} = +1$ are enforced using a protocol analogous to steps (i) and (ii) above.

B. Gauging the $\mathbb{Z}_2$ charge-conjugation symmetry

The $\mathbb{Z}_3$ toric code Hamiltonian possesses a global $\mathbb{Z}_2$ charge-conjugation symmetry, denoted $\prod \mathcal{C}$ (see Eqs. (S41) and (S55)). Gauging this global $\mathbb{Z}_2$ symmetry means promoting it to a local gauge symmetry, where an independent $\mathbb{Z}_2$ transformation can be performed locally at each "site". This procedure *enriches* the topological order, transforming the $\mathbb{Z}_3$ toric code into the $\mathcal{D}(S_3)$ quantum double model [S13, S14].

The gauging process can be implemented via an adaptive quantum circuit involving measurements and feed-forward, or via a purely unitary circuit composed of a deterministic sequence of gates. For the experiments presented in the main text, we prepare the $\mathcal{D}(S_3)$ ground state using a unitary protocol. The data for the $\mathcal{D}(S_3)$ ground state prepared using an adaptive circuit is shown in Fig. 2.

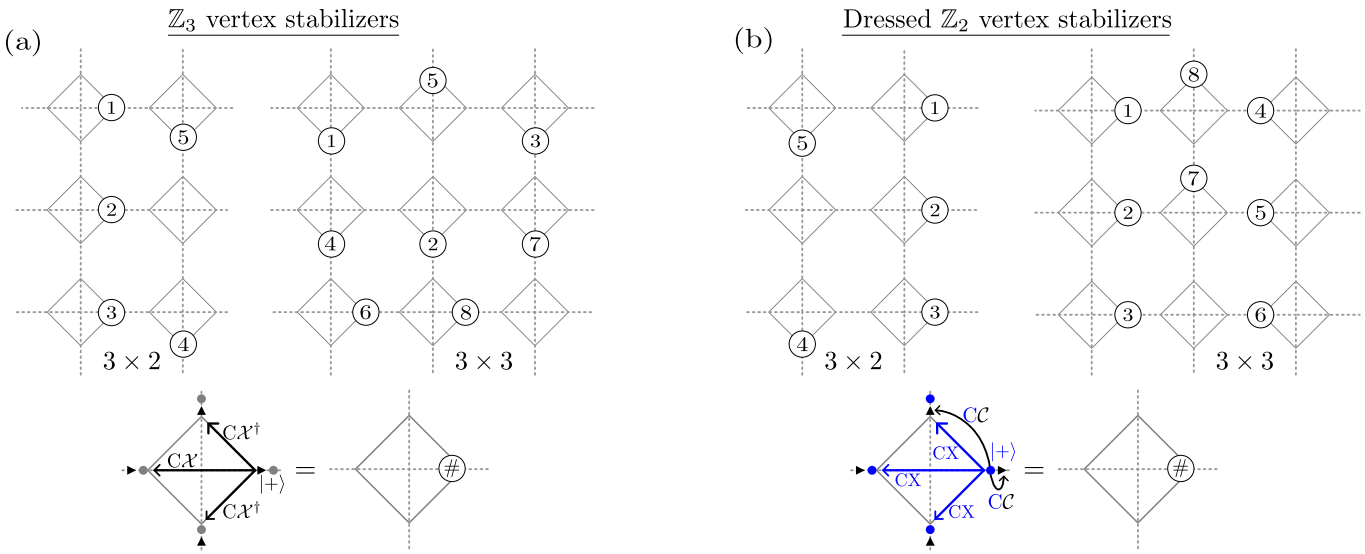
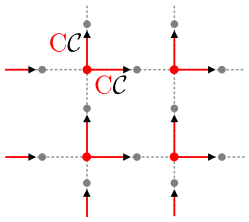


Figure 4. **Vertex Stabilizers Unitary Preparation.** Ordering for preparing vertex stabilizers in (a) the $\mathbb{Z}_3$ toric code and (b) the dressed $\mathbb{Z}_2$ toric code, for the 3×2 and 3×3 lattices. The dashed lines outline the square lattice. Rotated squares at each vertex represent the $S_{\mathcal{A}}$ stabilizers. Numbered edges denote the control qutrits, and the sequence of numbers indicates the order in which these vertex stabilizers are prepared.

1. *Using adaptive circuit*

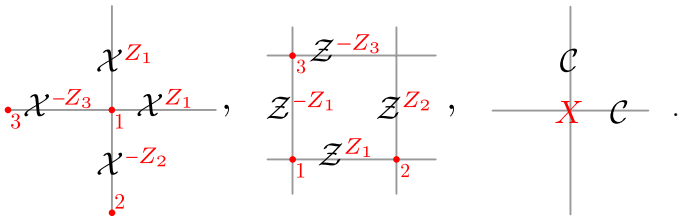
The gauging protocol augments the Hilbert space and enforces new local constraints:

1. Introduce an auxiliary qubit at each vertex of the lattice initialized in the $|+\rangle$ state, stabilized by X.
2. Apply a layer of $C\mathcal{C}$ gates that entangle the original $\mathbb{Z}_3$ qutrits with the new $\mathbb{Z}_2$ vertex qubits. Specifically, for each vertex, we apply two $C\mathcal{C}$ gates, where $\mathbb{Z}_2$ vertex qubit acts as the control and the right and top neighboring $\mathbb{Z}_3$ edge qutrits act as targets.



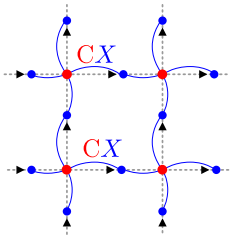
(S56)

Stabilizers of the joint state are now given by



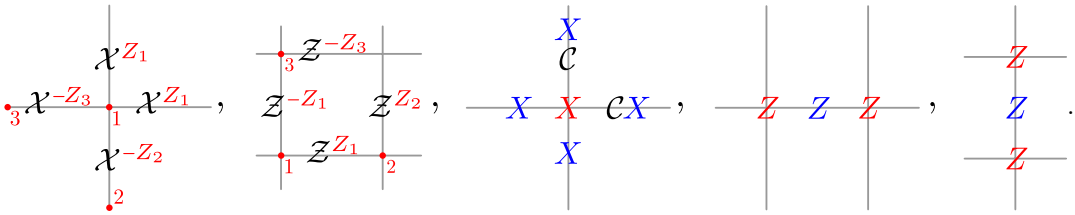
(S57)

3. We further add a qubit on each edge initialized in the $|0\rangle$ state, stabilized by Z. We apply CX gates between vertex and neighboring qubits with vertex qubits as the control and edge qubits as the target.



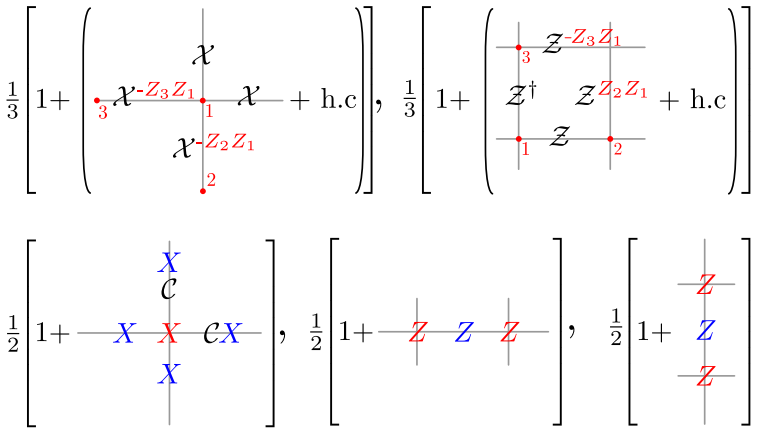
(S58)

The expanded set of stabilizers for the state augmented with edge qubits is given by



(S59)

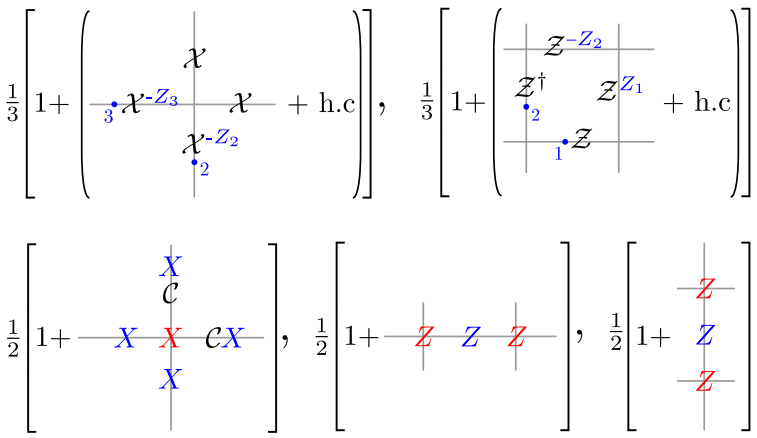
These stabilizers of the state do not all commute with each other. However, by writing these stabilizers as projectors over the trivial (+1) eigenspace, we obtain a set of commuting projectors with a *simpler* form (see [S13] for details)



(S60)

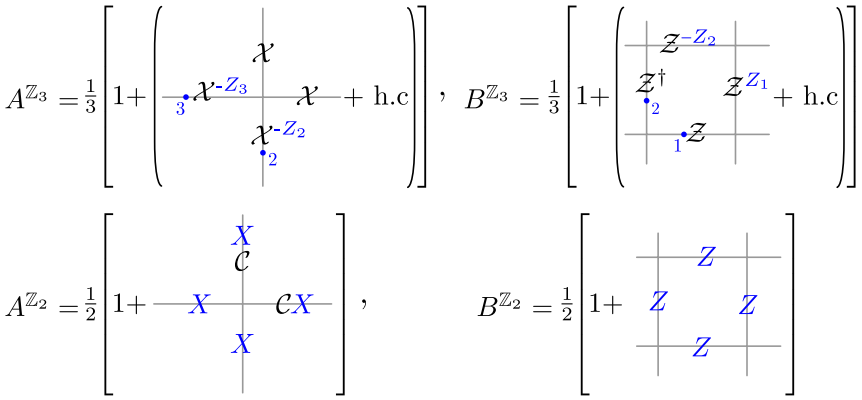
Moreover, the vertex qubit dependence of $\mathbb{Z}_3$ projectors can be eliminated by writing vertex qubit variables in

terms of edge qubits as determined by CX gates between vertex and edge qubits.



(S61)

4. Finally, we measure vertex qubits in X-basis and enforce local $\mathbb{Z}_2$ gauge invariance. This projects the state into the subspace where the original $\mathbb{Z}_3$ vertex constraints are now tied to the state of the $\mathbb{Z}_2$ gauge qubits. These measurements yield ± 1 outcomes. Based on the outcomes, we apply corrective Z's to the $\mathbb{Z}_2$ edge qubits. This feed-forward ensures that the system is deterministically prepared in the $+1$ eigenspace of all the projectors, namely

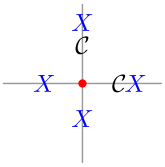


(S62)

The resulting projectors can be shown to be equivalent to the canonical quantum double projectors (see Section S2) within the ground state subspace of the $\mathcal{D}(S_3)$ model. In particular, the vertex projectors and the $B^{\mathbb{Z}_2}$ are exactly the same, whereas the $B^{\mathbb{Z}_3}$ given in Eq. (S34) in Section S2 can be reduced to the above expression when $B^{\mathbb{Z}_2} = +1$. While this method is optimal in terms of circuit depth, its reliance on mid-circuit measurements, feed-forward, and ancilla qubits affects the fidelity of the prepared state.

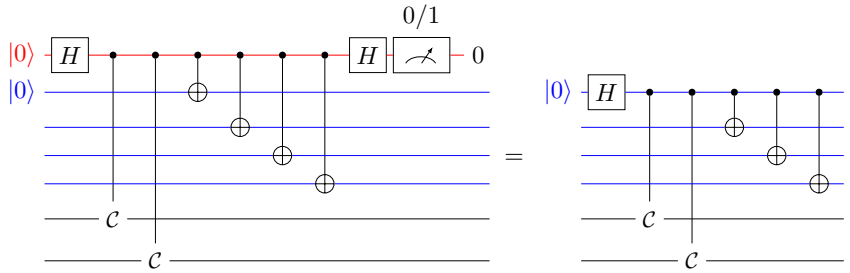
2. *Using unitary circuit*

We developed a purely unitary protocol that requires no measurements or feed-forward. The charge-conjugation gauging procedure, as described above, effectively amounts to measuring *dressed* vertex stabilizers of the $\mathbb{Z}_2$ toric code using an ancilla, after preparing the $\mathbb{Z}_3$ toric code, which is given as



(S63)

We can omit the need for introducing the ancilla (red vertex) qubit in the stabilizer measurement by using the following circuit identity.



(S64)

where blue and red lines correspond to qubits, and black lines correspond to qutrits. This identity shows that a circuit involving an ancilla measurement (left) can be replaced by a purely unitary sequence of gates (right), provided that one of the target qubits is initialized in the $|0\rangle$ state. This constraint forces the preparation of the dressed $\mathbb{Z}_2$ vertex stabilizers to be performed sequentially, similar to the unitary preparation of the $\mathbb{Z}_3$ toric code ground state in Section S6 A. We select a representative qubit, prepare it in the qubit $|+\rangle$ state, and use it as a control to entangle its neighbors. This process is repeated for all vertices in a specific order, shown in Fig. 4b, ensuring that the final state is the ground state of the $\mathcal{D}(S_3)$ model.

C. Adaptive vs unitary preparation protocol

1. *Fidelity estimate*

The overall fidelity in the Quantinuum's H2 hardware is upper-bounded by the fidelity of the native two-qubit gate (around 0.998), which is the dominant source of error. Therefore, a quick back-of-the-envelope estimation of the ground state preparation fidelity is to use the following formula

$$(\text{Fidelity per edge}) = (\text{Fidelity of two-qubit gate})^{\frac{d}{N}} \tag{S65}$$

where d is the total number of native two-qubit gates used in the preparation circuit, and $N = 3 \times 3 \times 2 = 18$ is the number of edges of the square lattice (for our 3×3 geometry).

For the unitary preparation protocol, the circuit depth is reported in the Supplementary Table S3, for the ground state unitary preparation, the native two-qubit gate depth $d = 238$. Therefore, the fidelity per edge is $(0.998)^{238/18} \approx 0.974$, which is within the fidelity bound of 0.970(5)–0.988(3) given in Eq. 3 in the main text.

For the measurement-based adaptive preparation protocol, the native two-qubit gate depth $d = 274$. Therefore, the fidelity per edge is $(0.998)^{274/18} \approx 0.970$, which is within the fidelity bound of 0.930(8)–0.978(2) given right below Eq. 3 in the main text. The fidelity is slightly worse than unitary preparation protocol because of the constant overhead involved for measurement-based protocol. Indeed, the measurement-based protocol requires slightly more gates since we need to entangle to ancillas (which we subsequently measure).

2. *Scalability*

For the measurement-based adaptive preparation protocol, the native two-qubit gate depth $d = 274$. Therefore, the fidelity per edge is $(0.998)^{274/18} \approx 0.970$, which is within the fidelity bound of 0.930(8)–0.978(2) given right below Eq. 3 in the main text. The fidelity is slightly worse than unitary preparation protocol because of the constant overhead involved for measurement-based protocol. Indeed, the measurement-based protocol requires slightly more gates since we need to entangle to ancillas (which we subsequently measure).

For the current system sizes, error contributions are dominated by gate count rather than runtime; therefore, the unitary preparation has (slightly) better fidelity. Naturally, as system size is scaled up, the runtime of the unitary preparation scales linearly with system size and will eventually form a bottleneck, such that the adaptive protocol then wins out. More precisely, for the adaptive protocol, the number of gates grows proportional to system size ($\sim L^2$), while the preparation time remains constant, such that the fidelity-per-qubit will remain constant (which can then be combined with quantum error correction if this local fidelity is beyond some stability threshold). In contrast, for the unitary preparation, while the number of gates also grows as $\sim L^2$, the preparation time scales as $\sim L$, leading to a fidelity $\sim e^{-aL^3}$, which means $e^{-aL} \sim e^{-a\sqrt{N}}$ per qubit.

S7. C_2 AND C_3 FLUX INSERTION AND SINGLE NON-ABELIAN ANYON

This section details the quantum circuits required to prepare the ground states of the $\mathcal{D}(S_3)$ model threaded with non-trivial topological flux. Specifically, we focus on the C_2 and C_3 flux sectors. Furthermore, we outline the steps for preparing a single C_3 non-Abelian anyon, a capability that is fundamentally impossible in Abelian topological phases.

To access the non-trivial C_2 and C_3 sectors, we employ the symmetry un-gauging and re-gauging strategy described in Ref. S7. Construction circuits are provided for a 3×3 lattice on a torus.

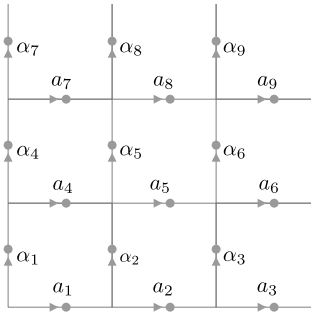


Figure 5. **3×3 square lattice with labeled edges**

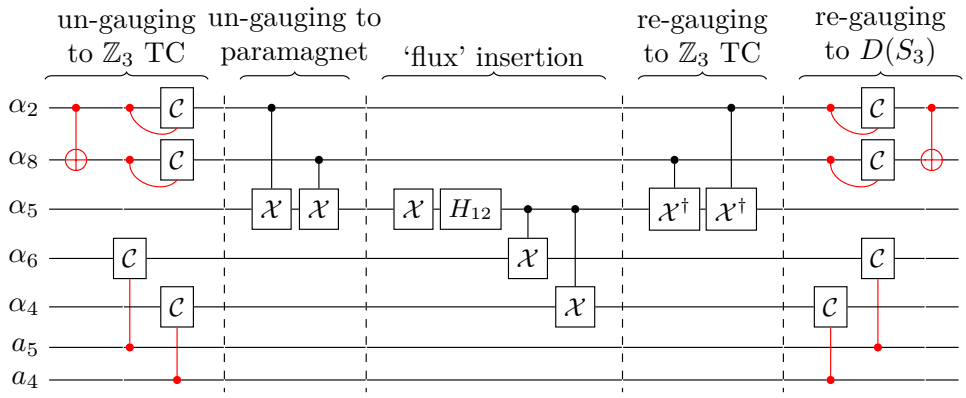
A. C_3 flux

The preparation of a ground state with C_3 flux, for instance $\left|\mathrm{gs}_{C_3,\mathrm{hori}}\right\rangle$, where the flux is threaded through bonds in the horizontal direction, involves:

- Ungauging the $\mathbb{Z}_2$ charge-conjugation symmetry.
- Ungauging the $\mathbb{Z}_3$ symmetry to access the 'paramagnet' qutrits.
- Applying a sequence of gates along a non-trivial horizontal loop to prepare the qutrits in the state $\frac{1}{\sqrt{2}}(|11\ldots\rangle_{\blacktriangle} + |22\ldots\rangle_{\blacktriangle})$.
- Re-gauging the $\mathbb{Z}_2$ and $\mathbb{Z}_3$ symmetries to return to the $\mathcal{D}(S_3)$ model with the desired C_3 flux.

The following is the optimized circuit for preparing $\left|\mathrm{gs}_{C_3,\mathrm{hori}}\right\rangle$ on a 3×3 lattice (see Fig. 5 for qudit labels) by applying it to the trivial ground state, $|\mathrm{gs}\rangle$.

$U_{C_3,\mathrm{hori}} =$



(S66)

In the circuits above, each black line represents a qubit–qutrit pair encoding an S_3 qudit. Gate actions on the qubit lines are shown in red, and the gates act as follows:

- CNOT acts on the qubit lines of the control and target qudits.
- C$\mathcal{X}$ acts on the qutrit lines of the control and target qudits.

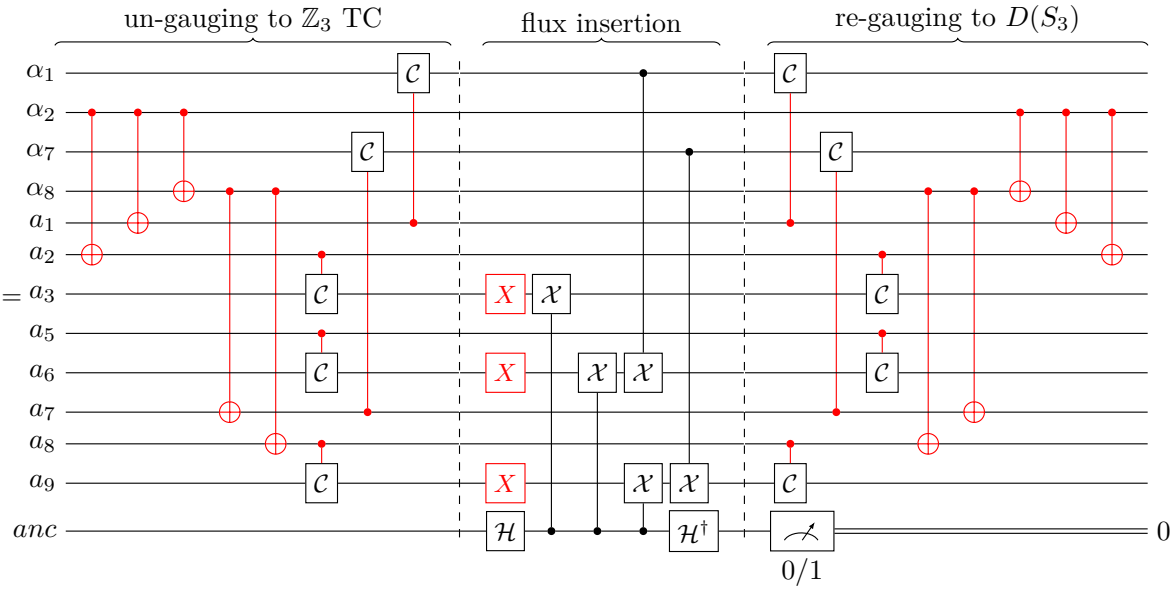
- C$\mathcal{C}$ acts on the qubit line of the control and qutrit line of the target qutrit. Hence, the "self-loop" C$\mathcal{C}$ gates at the un-gauging and re-gauging step act on distinct qubit and qutrit lines of a qudit.
- $\mathcal{X}$ and H_{12} act on the qutrit degrees of freedom of the qudits.

For detailed definitions and physical implementations, see Section S4.

B. C_2 flux

The construction of the ground state with C_2 flux requires ungauging to the $\mathbb{Z}_3$ toric code and then applying the projector $\frac{1}{3}(II\ldots+\mathcal{X}\mathcal{X}\ldots+\mathcal{X}^\dagger\mathcal{X}^\dagger\ldots)$ across a non-trivial loop on qutrits, , as well as X gates on the qubits. This requires an ancilla qutrit and complex feed-forward operations for deterministic preparation, as shown below, which creates the ground state $|\mathrm{gs}_{C_2,\mathrm{vert}}\rangle$ by acting on horizontal bonds across a non-trivial loop in the vertical direction.

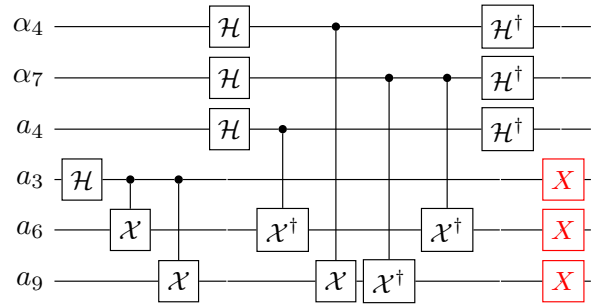
$U_{C_2,\mathrm{vert}} =$



(S67)

To simplify and avoid mid-circuits measurement, we constructed an optimized initialization circuit. The key insight is that the flux insertion operation in the above circuit can be commuted through the $\mathbb{Z}_3$ gauging unitary used in the standard $\mathcal{D}(S_3)$ state preparation. This shifts the flux-insertion step to an earlier stage, where it acts on a simple product state, thereby eliminating the need for an ancilla. The resulting initialization circuit,

$U^{\mathrm{init.}}_{C_2,\mathrm{vert}} =$



,

(S68)

is applied to the qudits *before* the main $\mathcal{D}(S_3)$ preparation unitary (see Section S4). This circuit effectively implements the +1 post-selected projector by using one of the qutrits as a control (i.e. a_4 in Eq. (S68)) for a series of C$\mathcal{X}$ gates. The final $\mathcal{D}(S_3)$ ground state prepared after this initialization has C_2 flux. Note that the correctness of this optimized initialization circuit depends on the ordering of plaquettes used during the $\mathbb{Z}_3$ toric code preparation (see Fig. 4).

C. Single C_3 flux anyon on the torus

Distinct, non-commuting flux sectors enable the creation of a single non-Abelian anyon through a mechanism unique to non-Abelian topological order. When a non-Abelian flux pair is created in a ground state that already carries a

different non-trivial flux, braiding between the two flux types changes the fusion channel of the pair. Consequently, fusing the pair does not return the system to the vacuum but instead leaves behind a single isolated anyon.

In our implementation, we begin from the ground state $|\mathrm{gs}_{C_2,\mathrm{vert}}\rangle$ (see Fig. 6a) and create a pair of C_3 fluxes that encircle the torus non-trivially along the horizontal direction (see Fig. 6b). Because the C_2 and C_3 fluxes braid non-trivially, the C_3 pair is forced out of the vacuum fusion channel. Upon fusion, a remnant C_3 non-Abelian anyon with quantum dimension 2 remains localized at the fusion site, consistent with the fusion rule

$$C_3 \times C_3 = [+] + [-] + C_3. \tag{S69}$$

This process can also be understood at the level of the $\mathbb{Z}_3$ toric code by ungauging the $\mathbb{Z}_2$ charge-conjugation symmetry. In this picture, the C_3 flux pair becomes the superposition of m and $\bar{m}$ anyons in the state $|m\rangle_1 |\bar{m}\rangle_2 + |\bar{m}\rangle_1 |m\rangle_2$, while the C_2 flux becomes a charge-conjugation defect [S15]. The C_2 non-contractible loop becomes a charge-conjugation domain wall, which toggles $m \leftrightarrow \bar{m}$. So moving the second particle of the pair across the charge-conjugation domain wall maps the state $|m\rangle_1 |\bar{m}\rangle_2 + |\bar{m}\rangle_1 |m\rangle_2 \to |m\rangle_1 |m\rangle_2 + |\bar{m}\rangle_1 |\bar{m}\rangle_2$. Finally, fusing particle 1 and 2 together gives $|m\rangle + |\bar{m}\rangle$; this is why the remnant particle is a C_3 flux. We note that the location of the single anyon excitation is where the fusion happened, not where the C_3 flux crosses the support of the $\hat{X}_{C_2}$ operator.

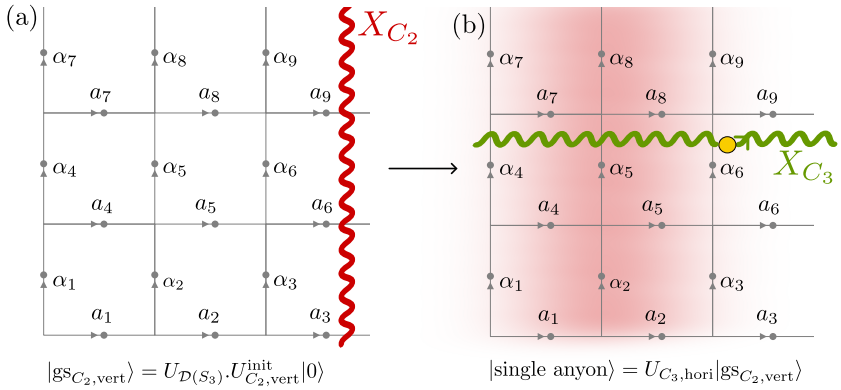


Figure 6. **Steps to create a single C_3 anyon**

We can also create a single non-Abelian [2] charge on the torus. One way to see it is by replacing the $m, \bar{m}$ pair with $e, \bar{e}$ pair while keeping the charge-conjugation domain wall the same. Running through the same argument as above, we obtain the state $|e\rangle + |\bar{e}\rangle$. Upon regauging the charge-conjugation symmetry, $|e\rangle, |\bar{e}\rangle$ becomes internal states of the [2] charge, thus we obtain a single [2] charge. Another more efficient perspective is that S_3 quantum double admits a $\mathbb{Z}_2$ anyon automorphism duality that interchange C_3 flux with [2] charge (this is inherited from the EM duality at the level of the qutrit toric code). Therefore by applying this duality transformation to the single C_3 flux state, we obtain the single [2] charge state.

S8. FIDELITY BOUNDS FOR $\mathcal{D}(S_3)$ GROUND STATE

This section establishes lower and upper bounds on the fidelity of the prepared state, ρ, with respect to the target ground state $|\mathrm{gs}\rangle$. The ground state is defined as the $+1$ eigenstate of the local vertex projectors $(A_v^{\mathbb{Z}_2}, A_v^{\mathbb{Z}_3})$ and local plaquette projectors $(B_p^{\mathbb{Z}_2}, B_p^{\mathbb{Z}_3})$. Furthermore, within the ground state manifold, $|\mathrm{gs}\rangle$ is uniquely identified by the $+1$ eigenspace of the horizontal and vertical non-contractable S_3 flux projectors, Z_{hori} and Z_{vert} (see Section S2). We can express the projector onto the ground state as:

$$|\mathrm{gs}\rangle\langle\mathrm{gs}| = \underbrace{\prod_v A_v^{\mathbb{Z}_2}}_{:=P_A^{\mathbb{Z}_2}} \cdot \underbrace{\prod_v A_v^{\mathbb{Z}_3}}_{:=P_A^{\mathbb{Z}_3}} \cdot \underbrace{\prod_p B_p^{\mathbb{Z}_2} \cdot \prod_p B_p^{\mathbb{Z}_3} \cdot Z_{\mathrm{hori}} \cdot Z_{\mathrm{vert}}}_{:=P_B} \tag{S70}$$

While all projectors commute, allowing for simultaneous measurement, we group them into 'product' projectors $P_A^{\mathbb{Z}_2}$, $P_A^{\mathbb{Z}_3}$, and P_B. These groupings are chosen to be as large as possible, subject to the constraint that their destructive measurement at the end of the circuit incurs minimal gate overhead.

For the experimentally prepared density matrix ρ, the fidelity with respect to the target state $|\text{gs}\rangle$ is:

$$\begin{aligned}\langle \text{gs}|\,\rho\,|\text{gs}\rangle &= \text{Tr}\left(\rho\,|\text{gs}\rangle\langle\text{gs}|\right)\\ &= \text{Tr}\left(\rho\; P_A^{\mathbb{Z}_2}\; P_A^{\mathbb{Z}_3}\; P_B\right)\end{aligned} \tag{S71}$$

We define the expectation values of the product projectors as:

$$\begin{aligned}p_A^{\mathbb{Z}_2} &:= \text{Tr}(\rho\; P_A^{\mathbb{Z}_2})\\ p_A^{\mathbb{Z}_3} &:= \text{Tr}(\rho\; P_A^{\mathbb{Z}_2})\\ p_B &:= \text{Tr}(\rho\; P_B)\end{aligned} \tag{S72}$$

To derive a lower bound for the fidelity, we use the following inequality which holds for a set of n commuting projectors $\{P_i\}$:

$$\prod_{i=1}^{n} P_i \geq \left(\sum_{i=1}^{n} P_i\right) - (n-1)\mathbb{I}. \tag{S73}$$

Taking the trace of both sides with the density matrix ρ, and applying Eqs. (S71) and (S72), we obtain:

$$\text{Tr}\left(\rho\; P_A^{\mathbb{Z}_2}\; P_A^{\mathbb{Z}_3}\; P_B\right) \geq\; (p_A^{\mathbb{Z}_2} + p_A^{\mathbb{Z}_B} + p_B - 2) \tag{S74}$$

Substituting the experimentally measured expectation values for the 3×3 lattice gives:

$$\langle \text{gs}|\,\rho\,|\text{gs}\rangle \geq 0.93(3) + 0.84(3) + 0.81(4) - 2 \tag{S75}$$

$$= 0.58(6), \tag{S76}$$

and the per-qudit fidelity is

$$f := \langle \text{gs}|\,\rho\,|\text{gs}\rangle^{1/18} \geq 0.970(5). \tag{S77}$$

Similarly, to get the upper bound for fidelity, we use the fact that the product of projectors is bounded by any individual projector:

$$\prod_{j=1}^{n} P_j \leq P_i \quad \forall i. \tag{S78}$$

Taking the trace with ρ yields:

$$\text{Tr}\left(\rho\; P_A^{\mathbb{Z}_2}\; P_A^{\mathbb{Z}_3}\; P_B\right) \leq \min\left\{p_A^{\mathbb{Z}_2},\; p_A^{\mathbb{Z}_B},\; p_B\right\}. \tag{S79}$$

Substituting the expectation values, we obtain:

$$\langle \text{gs}|\,\rho\,|\text{gs}\rangle \leq 0.81(4). \tag{S80}$$

In summary, the fidelity per qudit, f, for the ground state of the $\mathcal{D}(S_3)$ model on the 3×3 lattice is bounded by:

$$0.970(5) \leq f \leq 0.988(3). \tag{S81}$$

S9. BUREAU OF STANDARDS: THEORY

As touched upon in the main text, we are encoding a logical qutrit inside the zero-flux fusion subspace of two C_2 fluxes, defined by the following fusion rule:

$$C_2 \times C_2 = [+] + [2]. \tag{S82}$$

One qutrit basis state is supplied by the $[+]$ channel, and the other two are furnished by the $[2]$ channel—see Fig. 7(a). We note that this is a somewhat different logical encoding strategy that might be familiar to some readers, as we are

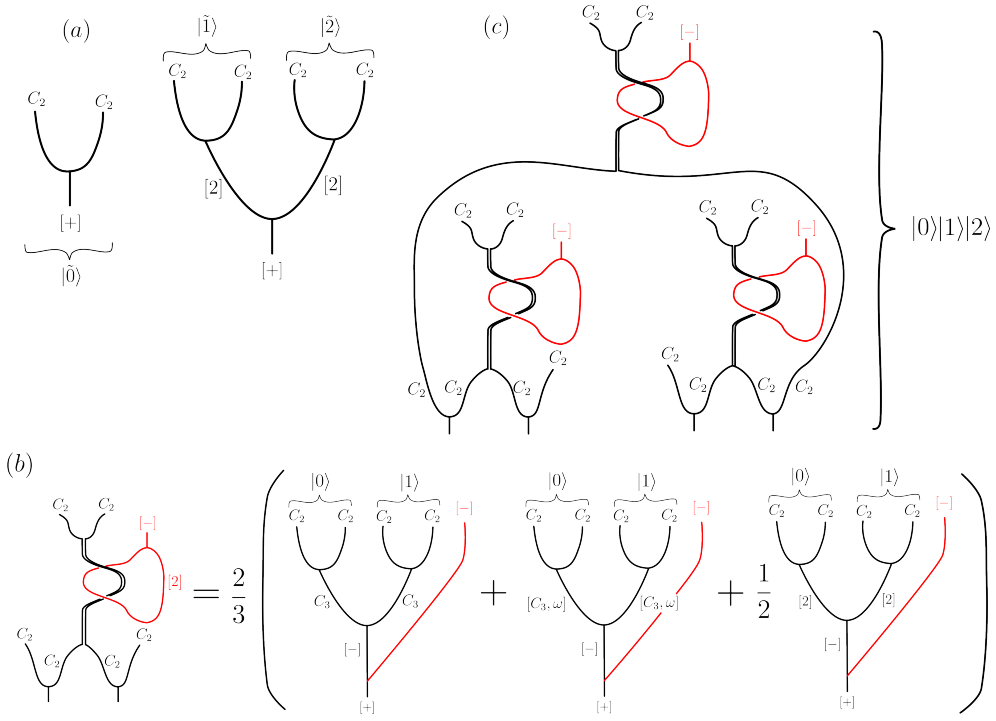


Figure 7. **Reformulating our approach as a fusion tree.** (a) Qutrit dual basis fusion trees; a $|\tilde{0}\rangle$ state is the vacuum state of two C_2 fluxes, while a reference pair of $|\tilde{1}\rangle$ and $|\tilde{2}\rangle$ state can be prepared by splitting a vacuum pair of [2] charges into four C_2 fluxes. Any further logical qubits will have to be compared with the $|\tilde{1}\rangle$ and $|\tilde{2}\rangle$ states to set their values in the dual basis. (b) The first step in setting up the computation basis bureau of standards. We assume the left pair is set as our reference $|0\rangle$, and we want to compare the right pair of fluxes in the computational basis. If the [2] charge braiding yields a remnant $[-]$, the right pair cannot be a $|0\rangle$, and so we can set it to be a $|1\rangle$. By re-writing the fusion and braiding diagram, we can express this $|0\rangle\,|1\rangle$ state in a more conventional fusion tree form, as a superposition over different intermediate fusion channels for the two logical flux pairs. (c) The fusion and braiding diagram corresponding to a full set of reference states, showing the three comparison measurements that ensure all three pairs of fluxes are in different computational basis states.

resolving the [2] charge outcome into its internal states. This type of encoding is sometimes known as a "split" fusion channel encoding [S4], to contrast with the usual fusion channel encoding where every anyon type has a single logical state associated with it.

The split fusion channel encoding treats the internal states of the [2] charge as distinct logical states: however, these internal states have no definite labeling, and are only physically meaningful as relative states. This subtlety necessitates the use of a "bureau of standards" which enables us to set a consistent convention of which state is which.

It is important to emphasize that, although we are using the internal states of the [2] charge as logical states, our encoding is *still topologically protected*, and cannot be accessed (and hence corrupted) locally. This is because the [2] charge is itself stored nonlocally between the C_2 fluxes that form the logical qutrit. Ultimately, this split fusion channel kind of encoding is equivalent to a more traditional fusion tree approach: we must simply take into account the reference states which form the bureau of standards, and the probe anyons that perform the comparison measurements, when writing down the fusion tree.

We build the bureau of standards for the Z-basis by starting with a pair of C_2 fluxes created from the vacuum. At this point, these fluxes must be in the $|\tilde{0}\rangle$ state:

$$|\tilde{0}\rangle = \frac{1}{\sqrt{3}} \left(|\sigma, \sigma\rangle + |\mu\sigma, \mu\sigma\rangle + |\bar{\mu}\sigma, \bar{\mu}\sigma\rangle \right). \tag{S83}$$

This initial pair of fluxes will act as our reference $|0\rangle$ state: any subsequent pair of fluxes must be entangled with this pair so that each branch of the wavefunction has a consistent convention. For instance, if we create a second pair of

fluxes to act as reference $|1\rangle$, we could entangle with the first "$|0\rangle$" in the following way:

$$\frac{1}{\sqrt{3}}(|\sigma,\sigma\rangle\,|\mu\sigma,\mu\sigma\rangle + |\mu\sigma,\mu\sigma\rangle\,|\bar{\mu}\sigma,\bar{\mu}\sigma\rangle + |\bar{\mu}\sigma\bar{\mu}\sigma\rangle\,|\sigma,\sigma\rangle). \tag{S84}$$

To create this entangled state, we first notice that if we take one flux from each pair, the combined flux of the two anyons will be C_3-valued. So we can start with our reference $|0\rangle$ state and a new vacuum pair of C_2 fluxes, performing a [2] charge braiding experiment around one half of each pair. If the [2] charge fuses to a remnant [−] charge, then the total flux encircled must have been a C_3, and we have projected into the desired state. In other words, we have performed a Z-basis comparison between the reference $|0\rangle$ and the new pair, found that they must be in different states, and so labeled the new pair as a $|1\rangle$.

This procedure can be expressed as a fusion and braiding diagram, see Fig. 7(b). Using the F- and R-symbol data for S_3, this diagram can be converted into a pure fusion diagram, demonstrating that the split fusion channel encoding can be understood as a usual fusion tree encoding with an expanded set of anyons. Fig. 7(c) shows the braiding and fusion diagram expressing the full bureau of standards protocol to create all three computational basis reference states, which can be simplified into a fusion diagram involving the three pairs of logical fluxes and the three remnant [−] charges.

S10. PROTOCOL IMPLEMENTATION DETAILS

A. Pull-through gate

1. *Theory prediction for local plaquette projector*

In Fig. 3 caption, we note that the theoretical prediction for the expectation value local plaquette projector $B^{\mathbb{Z}_3}$ is $\frac{1}{3}$. This is due to the fact that, for the local flux label, the C_2 flux we create has a equal and symmetric weight on all three labels: $\sigma, \mu\sigma, \bar{\mu}\sigma$. The $B^{\mathbb{Z}_3}$ projector projects to the sector where the qutrit value is 0, i.e. to the label σ. Therefore, the expectation value of $B^{\mathbb{Z}_3}$ is $\frac{1}{3}$.

2. *Theory prediction for nonlocal correlator*

In the caption of Fig. 3, we also note that the nonlocal correlator $\left\langle \Pi^{\mathbb{Z}_3}_{W_2W_3} \right\rangle$ has a predicted value of 1/3 at $t = 0$. The logical state is $\left|\tilde{0}\right\rangle_L |0\rangle_L = \frac{1}{\sqrt{3}}(|0\rangle_L + |1\rangle_L + |2\rangle_L)\,|0\rangle_L$. We see that there is equal weight between the $\mathcal{Z}$-basis eigenstates. Therefore, the expectation value of the nonlocal correlator

$$\left\langle \Pi^{\mathbb{Z}_3}_{W_2W_3} \right\rangle = \langle\tilde{0}|\,\langle 0|\,\Pi^{\mathbb{Z}_3}_{W_2W_3}\,|\tilde{0}\rangle\,|0\rangle = \left(\frac{1}{\sqrt{3}}\,\langle 0|\,\langle 0|\right)\left(\frac{1}{\sqrt{3}}\,|0\rangle\,|0\rangle\right) = \frac{1}{3}. \tag{S85}$$

At $t = 6$, the logical state is the qutrit Bell state $\frac{1}{\sqrt{3}}(|0\rangle_L\,|0\rangle_L + |1\rangle_L\,|2\rangle_L + |2\rangle_L\,|1\rangle_L)$. The $\Pi^{\mathbb{Z}_3}_{W_2W_3}$ correlator is designed to evaluate the $\mathbb{Z}_3$ part of the global flux and check whether it multiply to the trivial group element. For example, the state $|1\rangle_L\,|2\rangle_L$ has global flux $\mu\sigma$ and $\bar{\mu}\sigma$, such that the total qutrit flux is $\mu\cdot\bar{\mu} = e$; similarly for the $|2\rangle_L\,|1\rangle_L$ state. Therefore, the state is manifestly in the +1 eigenstate of the $\Pi^{\mathbb{Z}_3}_{W_2W_3}$ correlator.

3. *Circuits*

The system is on a 3×3 square lattice, where the indexing of qutrits (denoted by a_i, α_i) and qubits (denoted by b_i, β_i) on each edge is as follows:

4. *Step 1: Initialize control qutrit*

We initialize a pair of C_2 flux (at the (1,1) and (2,1) plaquettes) to be the control qutrit in the $\left|\tilde{0}\right\rangle_L$ state at $t = 0$ depicted in the main text Fig. 3. For the circuit below, the ancilla is initialized in the $|0\rangle_a$ state, such that the first qutrit $\mathcal{H}$ transforms the ancilla to be in the $\left|\tilde{0}\right\rangle_a$ state. Then, the flux's internal state is entangled with the ancilla via

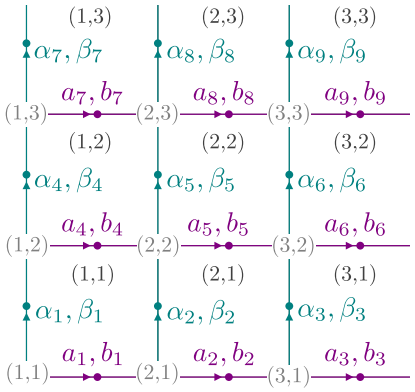
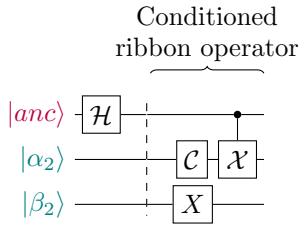
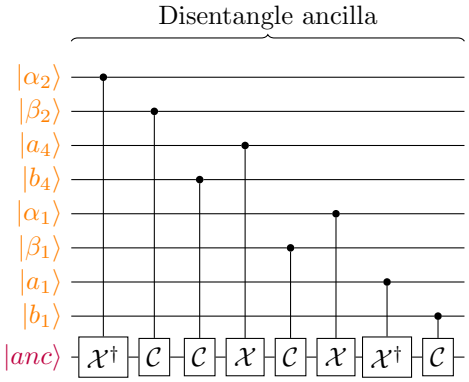


Figure 8. Labeling of vertices, edges, and plaquettes on a 3×3 square lattice on torus. The coordinate of vertex and plaquette are of the form (x, y) indexed from 1. The edges are indexed from left to right and bottom to top (alternating between horizontal and vertical edges).

action of ribbon operator with internal state conditioned on the ancilla state; we refer to this as state injection (see circuit below). This creates the state $|0\rangle_L |0\rangle_a + |1\rangle_L |1\rangle_a + |2\rangle_L |2\rangle_a$. The action is derived from the unitary circuit representation of C_2 ribbon operators (an example is shown in Eq. S21). Significant simplifications in the circuit (in particular, there are no ungauging and regauging layers) derives from the fact that there is no direct triangle in the ribbon.



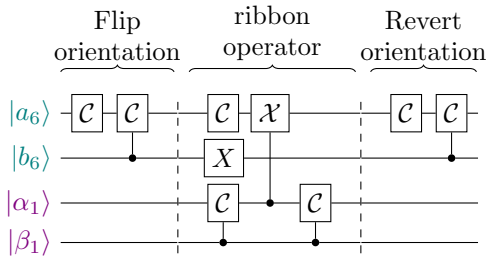
After creating the entangled state between the data qutrit and the ancilla, we need to disentangle the ancilla to initialize the data qutrit in the $\left|\tilde{1}\right\rangle_L$ state. To disentangle the ancilla, we need to apply a $C_L\mathcal{X}_a^\dagger$ gate so as to obtain the state $\left|\tilde{1}\right\rangle_L |0\rangle_a$. To control on the logical qutrit, we need to conditioned on the value of the flux loop (based at the origin) around the $(3, 1)$ plaquette. The following circuit implements the $C_L\mathcal{X}_a^\dagger$ gate to disentangles ancilla from the logical state.



5. *Initialize target qutrit*

We initialize a second C_2 flux pair (at the (3,1) and (3,2) plaquettes) to serve as the target qutrit in the $|0\rangle$ state at $t = 0$ depicted in the main text Fig. 3. As this is a $\mathcal{Z}$-basis eigenstate, no ancilla is needed to initialize the state. The

circuit is conjugated by gates that change the orientation of the edge for the qudits $|a_6, b_6\rangle$ such that the orientation of the triangle operator is aligned.



6. Coherent moving of flux example

Here, we provide an example circuit for moving flux coherently, when we move control qutrit from the (2,1) plaquette to the (3,3) plaquette as illustrated in Fig. 9. This is the step at $t = 1$ depicted in the main text Fig. 3. For all the remaining steps of the protocol (from $t = 2$ to $t = 6$), similar circuits are used to coherently move the fluxes to complete the braiding.

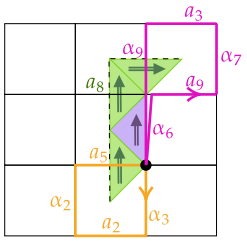
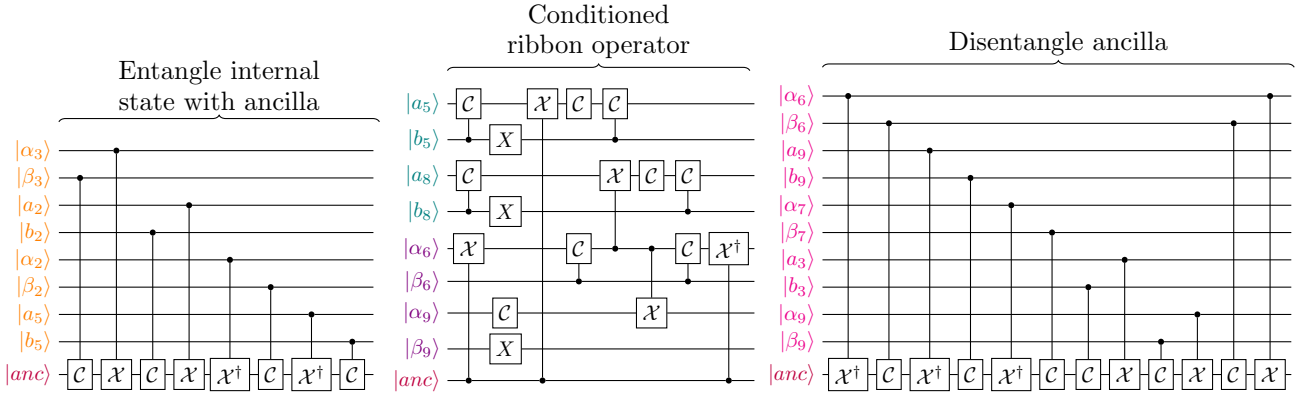


Figure 9. **Schematic for the pull-through step at $t = 1$, moving a flux of the control flux pair.**

First, the local flux internal state of the flux (measured in a clockwise loop depicted in orange based at the black dot) is entangled with the ancilla. Then, conditioned on the ancilla, a corresponding ribbon operator is applied to cancel out the flux at the initial plaquette. Finally, the ancilla is disentangled by applying a $C_L\mathcal{X}_a^\dagger$ gate (flux value measured in a counterclockwise loop depicted in magenta) *based at the same base point* (black dot). We emphasize that the base point has to be the same as the start of the ribbon to ensure that the ancilla is properly disentangled (see Section S3 for explanation).

The ancilla is initialized in the $|0\rangle$ state. The circuit on the left panel entangles the ancilla state with the internal local flux state of one of the control qutrit by implementing a $C_L\mathcal{X}_a$ gate. For the circuit in the middle panel, conditioned on the ancilla state, it applies the corresponding ribbon operator to cancel out the flux at the initial plaquette. Since the ribbon starts with a dual triangle, whose action is decoupled from the rest of the ribbon, its action is directly conditional on the ancilla state, just as the first direct triangle. Finally, for the circuit on the right panel, the ancilla is disentangled by implementing a $C_L\mathcal{X}_a^\dagger$ gate.



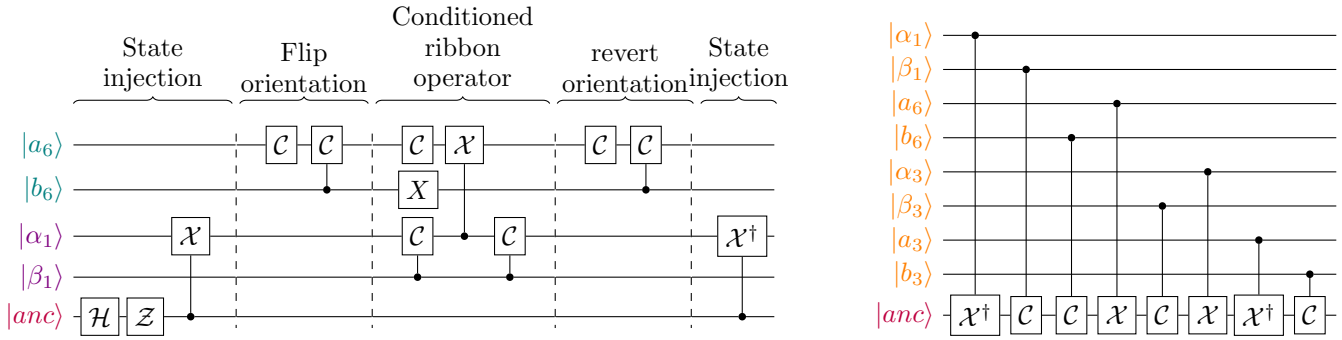
B. $\mathcal{X}$-basis measurement

Below, we detail the circuit for implementing the $\mathcal{X}$-basis measurement protocol for the case where the data logical qutrit is in the $|\tilde{1}\rangle$ state. For the case where the data logical qutrit is in the $|\tilde{0}\rangle$ state, the only difference is in step 1, where we omit the $\mathcal{Z}$ gate in the initial state injection stage; the rest of the circuit remains the same.

1. *Step 1: Create data logical* $|\tilde{1}\rangle$

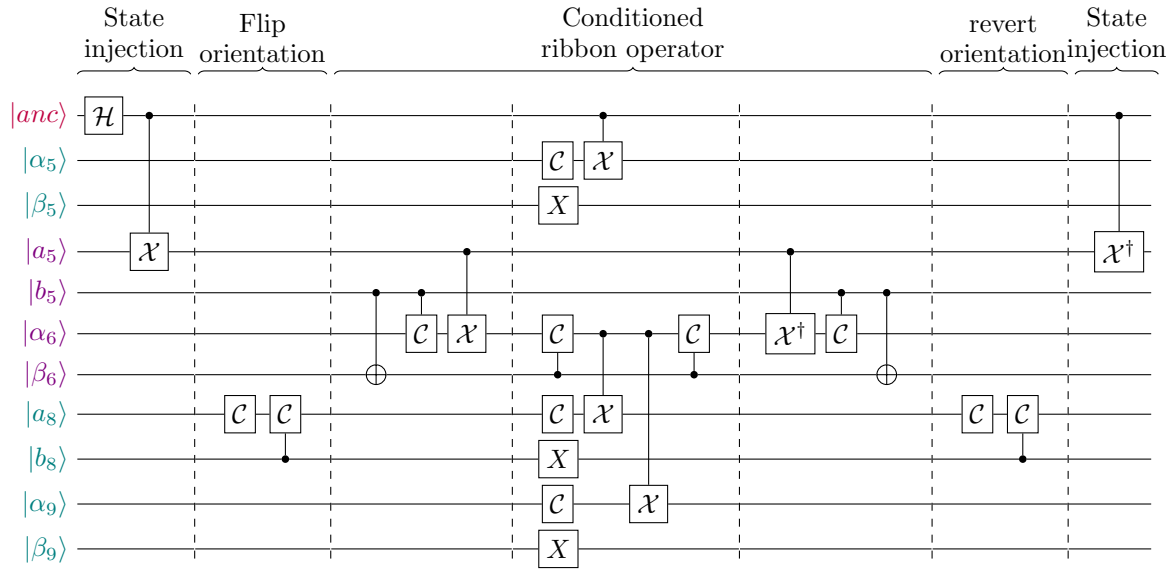
First, we create a C_2 flux pair at the $(3,1)$ and $(3,2)$ plaquettes (see Fig. 8 for labels); this flux pair serves as the data qutrit and is initialized in the $|\tilde{1}\rangle_L$ logical state by the end of step 1 using state injection similar to the procedure in the step 1 of pull-through gate (see Section S10 A 4). The circuit implementing the entangling is shown on the left panel, where the qudits in violet (teal) font are acted on by direct (dual) triangle of the ribbon operator.

The circuit on the right panel disentangle the ancilla by applying a $C_L\mathcal{X}_a^\dagger$ gate to obtain the (periodic boundary condition is used to shorten the flux loop).

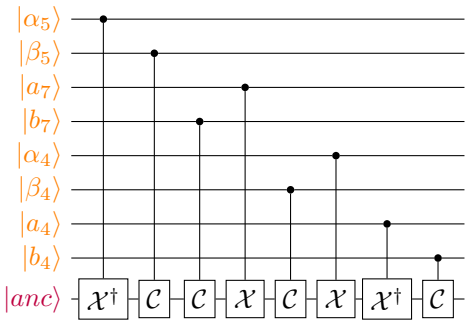


2. *Step 2: Create measurement pure flux*

We then create the second flux pair in the $|\tilde{0}\rangle_L$ logical state at the $(1,2)$ and $(3,3)$ plaquette, using similar techniques as in step 1.



Similar to the second part of step 1, we disentangle the ancilla from the logical state by applying a $C_L\mathcal{X}_a^\dagger$ gate, conditioned on the value of the flux loop (based at the origin) around the $(1,2)$ plaquette using periodic boundary condition.



3. *Step 3-5: Coherent moving fluxes*

We schematically illustrate the coherent moving circuits for step 3-5, similar to those implemented in Section S10 A 6.

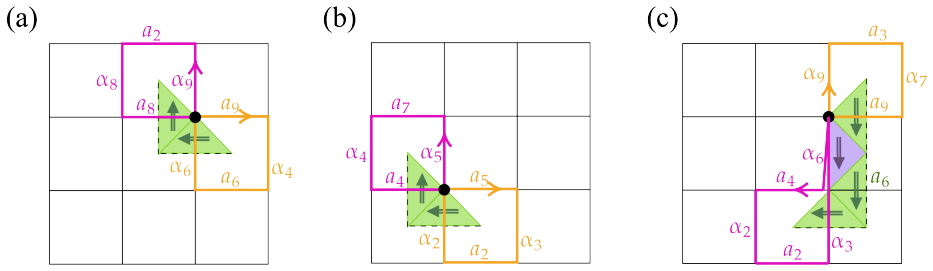
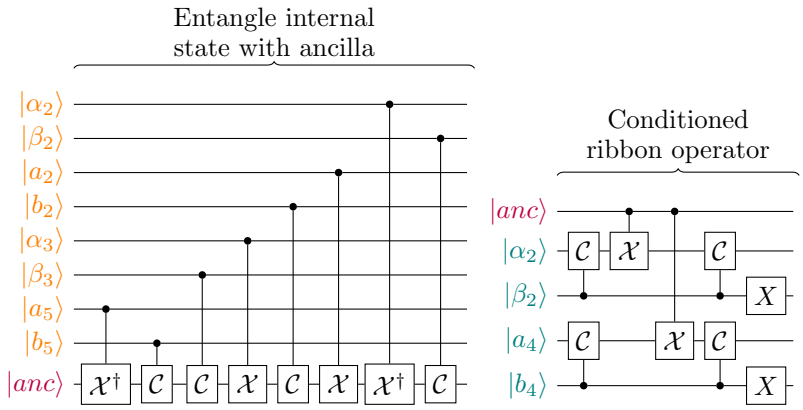


Figure 10. **Schematics for the coherent moving circuits for $\mathcal{X}$-basis measurement.** (a) Step 3, (b) step 4, and (c) step 5.

4. *Step 6: Fuse C_2 flux*

For the last step of the $\mathcal{X}$-basis measurement protocol, it involves fusing two C_2 flux, which warrants more explanation as the circuit towards the end is different from that of the coherent moving circuits in previous steps.

The entangling (left panel) and conditioned ribbon operator (middle panel) of the circuit proceeds in the same way as the coherent moving circuits, as shown below:



At this point, the flux pair have fused together and leaves behind a remnant charge at the fusion site. The ancilla is entangled with the system in a non-obvious basis. First, a qutrit Hadamard $\mathcal{H}^\dagger$ is applied, because this would reset the ancilla from the $|\tilde{0}\rangle$ state back to the $|0\rangle$ state if the ancilla is not entangled; the state is then in the basis where the internal state of the remnant charge is entangled with the ancilla as follows:

$$|0\rangle_a |+\rangle + |1\rangle_a |2-\rangle + |2\rangle_a |2+\rangle . \tag{S86}$$

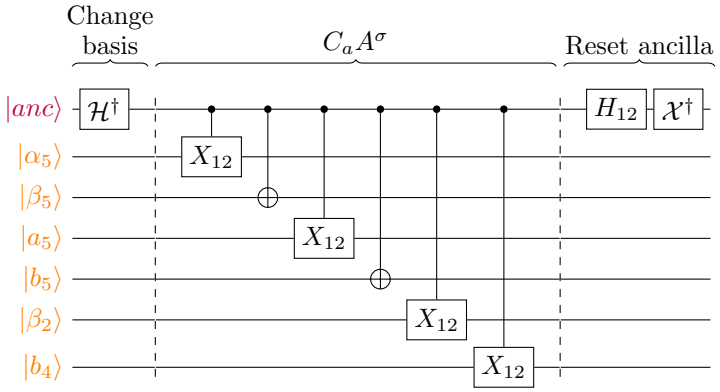
At this point, we could in principle end the protocol; projective measurement on the ancilla is equivalent to measuring the local charge at the fusion site. However, the ancilla is entangled with the system and cannot be reused until projective measurement.

We embrace a quixotic spirit by asking the question: can the ancilla be disentangled? Besides freeing the ancilla for other parts of the protocol, this also has the additional benefit of allowing the usual local vertex projectors ($A_v^{\mathbb{Z}_2}$ and $A_v^{\mathbb{Z}_3}$) to directly determine the charge violations.

Here, we make use of the fact that for the $\left|\tilde{1}\right\rangle_L$ input state of the $\mathcal{X}$-basis measurement protocol, there is no weight in the vacuum fusion channel, implying our state simplifies to

$$|1\rangle_a |2-\rangle + |2\rangle_a |2+\rangle . \tag{S87}$$

To disentangle the ancilla from the above state, we need to implement a special $C_a A^\sigma$ gate on the vertex of the fusion site: apply identity when the control is in $|1\rangle$ state, and apply A^σ when the control is in $|2\rangle$ state (since $A^\sigma |2-\rangle = |2+\rangle$). To decompose this further, the action of A^σ involves X gates on all edges of a vertex, and additional $\mathcal{C}$ gates on the oppositely aligned edges. The circuit implementation is as follows:



The definitions and physical implementations of the the qutrit–control qubit–target C_2NOT gate and the qutrit–control qutrit–target CX_{12} gate are given in Eqs. (S51) and (S52), respectively.

Finally, after applying the $C_{a_{12}} A^\sigma$ gate, the ancilla is disentangled and is in the $|1\rangle + |2\rangle$ state. Therefore, a H_{12} gate (defined as qubit Hadamard on the $\{|1\rangle, |2\rangle\}$ subspace in Eq. (S47)) followed by a $\mathcal{X}^\dagger$ are applied to revert the ancilla back to the $|0\rangle$ state.

C. $\mathcal{Z}$-basis measurement

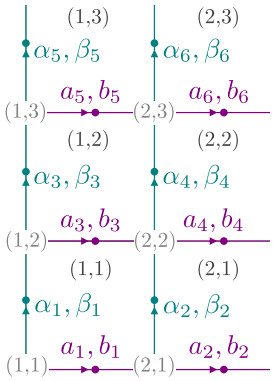
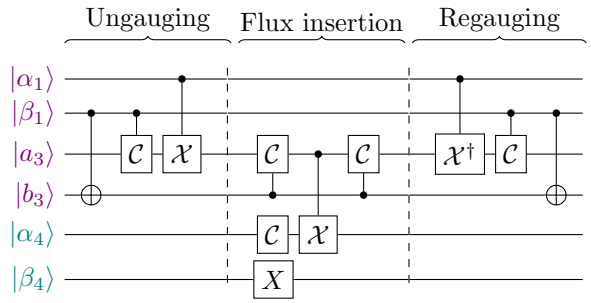


Figure 11. Labeling of vertices, edges, and plaquettes on a 3×2 periodic square lattice. Vertex and plaquette coordinates are (x, y), indexed from 1. Edges are ordered left to right and bottom to top, alternating between horizontal and vertical.

1. *Example of generalized ribbon operator*

Here, we provide an example of how the generalized ribbon operator (see Section S1 D 3) is used to specify a logical qutrit in the absolute encoding IV A. It involves prepending the usual ribbon with a "direct string" that connects from the starting vertex of the ribbon to the origin. Operationally, this means that the ungauging (regauging) step of the ribbon operator starts (ends) at the origin.

As an example, we show a circuit implementation of step 1 of the $\mathcal{Z}$-basis measurement protocol IV D, where we create the data qutrit in the $|0\rangle_L$ state at the (1,2) and (2,2) plaquettes:

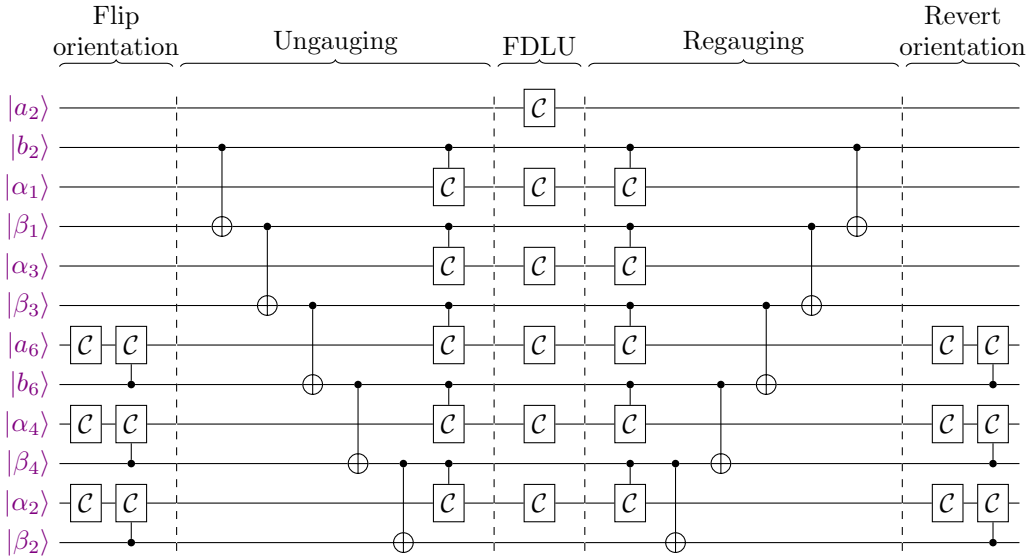


We can decompose the above circuit into direct string and the following triangle operators in two ways: 1) the direct string has support on the (α_1, β_1) and (a_3, b_3) edges, and the first triangle operator is the dual triangle acting on the (α_4, β_4) edge 2) the direct string has support on just the (α_1, β_1) edge, and the following is a ribbon consisting of two triangle operators (a direct triangle acting on the (a_3, b_3) edge and a dual triangle acting on the (α_4, β_4) edge). In either decomposition, the direct string is necessary, since there is no way to express the above circuit using only consecutive triangle operators.

Similar circuits are used for creating logical states in Section IV E and V.

2. [2] *charge braiding*

When we braid [2] charges around the two fluxes for the $\mathcal{Z}$-basis comparison measurement, we can perform it in one shot, i.e. implementing a single closed ribbon operator without the need for coherent moving. An example circuit is given below, at the step where we compare the data qutrit with the reference $|0\rangle_L$ qutrit:



It is slightly modified from the [2] charge unitary ribbon given in Eq. S20. The orientation of some direct edges are flipped at first, such that all direct edges' orientation is aligned with respect to the direction of the ribbon operator. The orientation is reverted after the application of the unitary ribbon.

Because of the nontrivial braiding of the [2] charge with a C_3 flux (e.g. when performing comparison measurement between the data qutrit and the reference $|1\rangle_L$ state), a closed loop ribbon operator is not "rotationally-invariant" along the loop. The start and end point of the ribbon operator determines the location of any remnant charge (in this case, a possible remnant [−] charge) that breaks this rotational invariance.

Similar circuits are used for $\mathcal{Z}$ comparison measurements implemented in Section IV E and V.

D. $\mathcal{Z}$ bureau of standards

The circuits used in $\mathcal{Z}$ bureau of standards are very similar to that used in $\mathcal{Z}$-basis measurement (see Section S10 C). The main difference is that the initial states are all in the $|0\rangle_L$ state (an example circuit creating such state is given in Section S10 B 2. Subsequent [2] charge braiding circuits are adapted from the example given in Section S10 C 2.

1. *Multi-verse interpretation of relative encoding*

First, we outline an argument by Mochon [S16] that illustrates how we can interpret the relative encoding as the multi-verse of possible flux labels, such that the absolute encoding is one "branch" of this multi-verse. Consider the state $|\tilde{0}\rangle_L = \frac{1}{\sqrt{3}}(|0\rangle + |1\rangle + |2\rangle)$. If we forget about the coherence between the $\mathcal{Z}$ eigenstates, it is a completely mixed state $\rho = \frac{1}{3}(|0\rangle\langle 0| + |1\rangle\langle 1| + |2\rangle\langle 2|)$. Equivalently, we can obtain this mixed state ρ by starting with a qutrit Bell state $|0\rangle|0\rangle + |1\rangle|1\rangle + |2\rangle|2\rangle$ and trace out the second qutrit. Then, we can use it as a $\mathcal{Z}$-basis reference state. Without loss of generality, we can assert it to be the $|0\rangle_L$ in the relative encoding. Then, other $\mathcal{Z}$ states can be instantiated by $\mathcal{Z}$-basis comparison measurement with the reference state.

For example, given a reference $|0\rangle_L$ in the relative encoding, if we instantiate another $|0\rangle_L$ state, this is actually the state $\psi \sim |0\rangle|0\rangle + |1\rangle|1\rangle + |2\rangle|2\rangle$ in the absolute encoding. What matters is that the logical qutrit values are the same for every "branch" of this state.

2. [2] *charge braiding once suffices for instantiating $\mathcal{Z}$ reference states*

Next, we explain why we only perform [2] charge braiding once rather than twice to instantiate the $\mathcal{Z}$ bureau of standards reference states.

For a full $\mathcal{Z}$-basis measurement, we need to braid a [2] charge pair twice to ensure that we obtain the state with the correct coherence in the orthogonal subspace after the measurement. For example, in Section V, two [2] charge braiding are implemented to ensure that after projecting out of the $|2\rangle_L$ subspace, the resulting state $|0\rangle_L + \omega|1\rangle_L$ has the correct relative phase factor.

However, for the $\mathcal{Z}$ bureau of standards protocol (Section IV E), we do not require performing such [2] charge braiding twice. This is because the coherence between the different "branches" of the relative encoding is not important; all that matters is that the absolute flux label of the states are distinct pair-wise for the three reference states. Hence, we do not need to repeat braiding with the [2] charge twice to obtain the correct logical reference states for the $\mathcal{Z}$ bureau of standards.

E. Magic state creation

The circuits used in creating the magic state $(|0\rangle + \omega|1\rangle)$ are very similar to that used in $\mathcal{Z}$-basis measurement (see Section S10 C). The main difference is that the initial states are in the logical $|2\rangle_L$ and $|\tilde{1}\rangle_L$ state. To create the logical $|2\rangle_L$ state, it is a slight modification of the circuit given in S10 C 1: the circuit is conjugated with $\mathcal{X}^\dagger$ (see remark on C_2 ribbon in Section S1 D 2 for explanation). For the logical $|\tilde{1}\rangle_L$ state, an example circuit is given in S10 B 1. Subsequent [2] charge braiding circuits are adapted from the example given in Section S10 C 2.

[S1] S. X. Cui, Topological quantum computation (2018).
[S2] C. F. B. Lo, A. Lyons, R. Verresen, A. Vishwanath, and N. Tantivasadakarn, Universal Quantum Computation with the S_3 Quantum Double: A Pedagogical Exposition (2025).
[S3] A. Kitaev, Annals of Physics **303**, 2 (2003).
[S4] S. X. Cui, S.-M. Hong, and Z. Wang, Quantum Information Processing **14**, 2687 (2015), arXiv:1401.7096 [quant-ph].
[S5] L. Chen, Y. Ren, R. Fan, and A. Jaffe, npj Quantum Information **11**, 112 (2025).
[S6] B. Yan, P. Chen, and S. X. Cui, Journal of Physics A: Mathematical and Theoretical **55**, 185201 (2022).
[S7] A. Lyons, C. F. B. Lo, N. Tantivasadakarn, A. Vishwanath, and R. Verresen, Phys. Rev. Lett. **135**, 200405 (2025).
[S8] N. Tantivasadakarn, W. Ji, and S. Vijay, Phys. Rev. B **104**, 115117 (2021).
[S9] C. F. B. Lo et al., Forthcoming.
[S10] Quantinuum System Model H2 Product Data Sheet, `https://www.quantinuum.com/products-solutions/system-model-h2-series`, (2024).
[S11] Quantinuum Hardware Specifications (2024).
[S12] S. Sivarajah, S. Dilkes, A. Cowtan, W. Simmons, A. Edgington, and R. Duncan, Quantum Science and Technology **6**, 014003 (2020).
[S13] R. Verresen, N. Tantivasadakarn, and A. Vishwanath, arXiv preprint arXiv:2112.03061 10.48550/arXiv.2112.03061 (2021).
[S14] N. Tantivasadakarn, A. Vishwanath, and R. Verresen, PRX Quantum **4**, 020339 (2023).
[S15] M. Iqbal, A. Lyons, C. F. B. Lo, N. Tantivasadakarn, J. Dreiling, C. Foltz, T. M. Gatterman, D. Gresh, N. Hewitt, C. A. Holliman, J. Johansen, B. Neyenhuis, Y. Matsuoka, M. Mills, S. A. Moses, P. Siegfried, A. Vishwanath, R. Verresen, and H. Dreyer, Nature Communications **16**, 6301 (2025).
[S16] C. Mochon, Phys. Rev. A **69**, 032306 (2004).