Demonstration of High-Fidelity Entangled Logical Qubits using Transmons

Arian Vezvaee, ^{1,2,3,*} Vinay Tripathi, ^{2,4,†} Mario Morford-Oberst, ^{1,2} Friederike Butt, ^{5,6} Victor Kasatkin, ^{1,2} and Daniel A. Lidar^{1,2,3,4,7}

¹Department of Electrical & Computer Engineering, University of Southern California, Los Angeles, California 90089, USA ²Center for Quantum Information Science & Technology, University of Southern California, Los Angeles, CA 90089, USA ³Quantum Elements, Inc.

⁴Department of Physics & Astronomy, University of Southern California, Los Angeles, California 90089, USA

⁵Institute for Quantum Information, RWTH Aachen University, Aachen, Germany

⁶Institute for Theoretical Nanoelectronics (PGI-2), Forschungszentrum Jülich, Jülich, Germany

⁷Department of Chemistry, University of Southern California, Los Angeles, California 90089, USA

Quantum error correction (QEC) codes are necessary to fault-tolerantly operate quantum computers. However, every such code is inherently limited by its inability to detect logical errors. Here, we propose and implement a method that leverages dynamical decoupling (DD) to drastically suppress logical errors. The key to achieving this is to use the logical operators of the QEC code as DD pulses, which we refer to as logical dynamical decoupling (LDD). The resulting hybrid QEC-LDD strategy is in principle capable of handling arbitrary weight errors. We test QEC-LDD using IBM transmon devices and the [[4,2,2]] code, demonstrating performance that significantly exceeds the capabilities of using either this code or DD in isolation. We present a method that allows for the detection of logical errors affecting logically encoded Bell states, which, in this case, arise primarily from crosstalk among physical qubits. Building on this, we experimentally demonstrate high-fidelity entangled logical qubits.

Quantum error correction (QEC) [1–4] is fundamental to the realization of fault-tolerant quantum computation [5–8], ensuring the preservation of quantum information undergoing error processes during computation and storage [9, 10]. Numerous successful experimental demonstrations of QEC across various platforms have been reported over the years [11–15], with the scale and pace accelerating recently towards genuine fault tolerance [16–24].

Fault-tolerant quantum computing will require the stability of logical qubits over the potentially hours-long timescales of quantum algorithms solving utility-scale problems [25–27]. Above threshold, over such timescales low-weight physical errors may transform into logical errors of weight higher than any deployed fixed-distance QEC code can handle. This can become a concern even below threshold since long-range spatial and temporal correlations may develop that violate the assumptions underlying fault tolerance theory [28–32].

Conventionally, suppressing higher-weight errors requires increasing the QEC code distance, e.g., by means of code concatenation [33, 34], or increasingly larger codes such as is done with surface codes [35, 36], color codes [37], or quantum low-density parity-check codes [38]. Although effective, these strategies result in significant overhead, substantially increasing the number of physical qubits required and the time required to decode and correct errors.

Here, we propose and demonstrate a low-cost method

that combines a fixed-distance QEC or quantum error detection (QED) code with dynamical decoupling (DD) [39–42]. This hybrid method, which we call QEC-LDD, can handle arbitrary-weight errors.

Here, LDD stands for 'logical dynamical decoupling'; the decoupling sequence is implemented using the logical operators of the QEC code as pulses.

We prove a corresponding QEC-LDD theorem which, informally, posits that the most general set of errors acting on an [n, k, d] code, namely errors with weights up to n, can all either be suppressed using LDD or corrected using the code, for any $d \ge 1$. This seemingly paradoxical result is enabled by virtue of the fact that the LDD sequence suppresses all logical errors as well as some physical errors, thus leaving a much smaller subset of errors for the code to handle, which turn out to all be correctable. Notably, these are not the errors the code was originally designed to correct; indeed, the result holds in particular also for purely error detecting codes. The QEC-LDD theorem is a formal result, in the sense that QEC-LDD, if implemented literally as described in the theorem, is not expected to outperform a pure DD protocol (absent any QEC) on k physical qubits. Nevertheless, the QEC-LDD theorem serves as a useful framework for less ambitious protocols that do not attempt to remove all errors but instead focus on the more physically relevant subset. We discuss and illustrate this conclusion in detail in this work, in the context of a transmon-based demonstration of entangled logical qubits.

Standard, 'physical' DD, where DD pulses act not at the logical level but at the physical qubit level, has recently shown great progress, improving the fidelity of storing quantum states [43–49], circuits [50–

^{*} vezvaee@usc.edu

[†] current affiliation: IBM Quantum

52], and even the performance of entire algorithms [53–56]. Furthermore, physical DD can be seamlessly combined with fault-tolerant QEC [57, 58] and several recent QEC experiments have used physical DD profitably [18–21, 59, 60].

However, such pulse sequences can introduce additional errors due to control imperfections in the pulses and due to crosstalk, potentially overshadowing the benefits of DD. To address this challenge, we design our LDD sequences to be robust against such control errors [61, 62] and crosstalk [63–67], ensuring that the advantages of logical error suppression are not compromised. This robust design allows our LDD implementation to enhance the protection of the code space as intended, effectively targeting both logical and control errors. We demonstrate the practical utility and advantages of this approach using 127-qubit IBM quantum processors.

The workhorse in our demonstrations is the [4, 2, 2]quantum error detection code, whose two logical qubits we use to prepare logically encoded Bell states. demonstrate the effectiveness of LDD in suppressing logical errors, we first need a method to detect such errors. However, this code is constrained by its low distance and is incapable of detecting prevalent logical errors, including ZZ crosstalk errors. To overcome this limitation, we design experiments in which, through the use of logically encoded Bell states, we nevertheless unequivocally detect the occurrence of logical errors for a known input logical state. This is then followed by the implementation of various versions of LDD, each employing different subsets of logical operators to construct the sequence. These implementations effectively demonstrate significant suppression of logical errors and substantial improvement in the fidelity of the code space.

The structure of this paper is as follows. In Section I we provide pertinent background on the error model and DD from the perspective of group averaging. We discuss standard Pauli-group DD as well as DD implemented using the stabilizers and or logical operators of a stabilizer code. In Section II we state and prove the QEC-LDD Theorem. We also illustrate the theorem using the example of the [[4,2,2]] code. We then turn to our experimental demonstration of high-fidelity logical Bell states in Section III. This section describes our experimental design and methodology, evidence that both physical DD and error detection with postselection improve logical Bell state fidelity, and finally the evidence that the hybrid QEC-LDD strategy is significantly better.

We conclude in Section $\overline{\text{IV}}$ and provide supporting technical details in the Appendix.

I. BACKGROUND

A. Quantum codes and logical errors

An [[n, k, d]] code encodes k logical qubits into n physical qubits with distance d [68]. The weight of an error is

the number of physical qubits it affects simultaneously. The highest-weight errors an [n, k, d] code can correct or detect are $\lfloor \frac{d-1}{2} \rfloor$ and d-1, respectively. A d=2 code is a pure quantum error detection (QED) code, and a distance-d QEC code can always be used as a QED code for errors of weight < d. The code distance is the minimum number of physical qubits that must experience an error to cause an undetectable logical error, i.e., an error forming a logical operation inside the code space. These logical errors can either be inherently present or result from the accumulation of lower-weight errors over time. For example, in the context of superconducting qubits with fixed-frequency couplers, a prevalent challenge is ZZ crosstalk [69, 70], which inherently introduces weight-two errors that can present as logical errors for distance-2 codes.

B. Logical dynamical decoupling

Let $\mathcal{P}_n = \{\pm P_j, \pm i P_j\}_{j=0}^{4^n-1} \subset \mathsf{U}(2^n)$ be the full Pauli group with 4^{n+1} elements generated by $\{I, X, Y, Z\}^{\otimes n}$ and the phase factors $\{\pm 1, \pm i\}$. Let $\tilde{\mathcal{P}}_n = \mathcal{P}_n/\{\pm 1, \pm i\} \subset \mathsf{U}(2^n)/\mathsf{U}(1)$ be the Abelian factor group with 4^n elements (here $\mathsf{U}(m)$ is the group of unitary operators on \mathbb{C}^m).

Consider the most general 'total decoherence' systembath interaction for n qubits:

$$H_{SB} = \sum_{j=0}^{4^n - 1} \alpha_j P_j \otimes B_j. \tag{1}$$

We let $P_0 = I^{\otimes n}$, the *n*-qubit identity operator. The pure-bath Hamiltonian $P_0 \otimes H_B$ is harmless.

In its simplest form, accounting only for first-order decoupling and ignoring pulse errors, DD theory can be understood as group symmetrization [71, 72]. Choose a decoupling group $\tilde{\mathcal{G}}$ that consists of a set of unitary transformations (pulses) g_j acting purely on the system: $\tilde{\mathcal{G}} = \{g_j\}_{j=0}^{|\tilde{\mathcal{G}}|-1}$, where $g_0 = P_0$. We assume that each such pulse is a unitary operator defined up to a global phase, i.e. $\tilde{\mathcal{G}} \subset \mathsf{U}(2^n)/\mathsf{U}(1)$; however, to simplify the notation we write, e.g., $g_0 = P_0$ instead of $g_0 = [P_0]$ (the equivalence class of P_0 up to a global phase). Similarly to \mathcal{P}_n and $\tilde{\mathcal{P}}_n$, we assume that $\tilde{\mathcal{G}}$ can be written as $\tilde{\mathcal{G}} = \mathcal{G}/\{\pm 1, \pm i\}$ for some group $\mathcal{G} \subset \mathsf{U}(2^n)$. In particular, we assume that the pulses are instantaneous and cycle over all elements of the group $\tilde{\mathcal{G}}$. This yields the DD pulse sequence:

$$U(T) = \prod_{j=0}^{|\tilde{\mathcal{G}}|-1} g_j^{\dagger} f_{\tau} g_j = e^{-iT(\langle H_{SB} \rangle_{\mathcal{G}} + P_0 \otimes H_B)} + O(T^2).$$
(2)

Here, τ is the pulse interval, $T = |\tilde{\mathcal{G}}|\tau$ is the total duration of one repetition of the sequence, $f_{\tau} = e^{-i\tau H_{SB}}$

is the free-evolution unitary, and $\langle H_{SB} \rangle_{\mathcal{G}}$ is the group-symmetrized system-bath Hamiltonian, where

$$\langle A \rangle_{\mathcal{G}} \equiv \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} g^{\dagger} A g = \frac{1}{|\tilde{\mathcal{G}}|} \sum_{g \in \tilde{\mathcal{G}}} g^{\dagger} A g$$
 (3)

is the projection of the arbitrary bounded system-bath operator A onto the subalgebra of operators that commute with every element of \mathcal{G} (i.e., its commutant). We say that $\tilde{\mathcal{G}}$ decouples H_{SB} (to first order) if $\langle H_{SB} \rangle_{\mathcal{G}} = cI$ where c is a constant, including zero.

For example, $\tilde{\mathcal{G}} = \tilde{\mathcal{P}}_n$ decouples an arbitrary n-qubit system-bath Hamiltonian since then $\langle H_{SB} \rangle_{\mathcal{G}} = 0$, albeit at a sequence time cost of $T = 4^n \tau$ [40]. To illustrate this, consider n = 1; the most general system-bath Hamiltonian of a single qubit is $H_{SB} = \sum_{i=0}^{3} \alpha_i P_i \otimes B_i$ with $P_i \in \{I, X, Y, Z\}$. If $\tilde{\mathcal{G}} = \{I, X, Y, Z\}$, then U(T) simplifies into the well-known XY4 sequence $U(4\tau) = Yf_{\tau}Xf_{\tau}Yf_{\tau}Xf_{\tau}$ [73], and indeed, $\langle H_{SB}\rangle_{\mathcal{G}} = 0$.

Higher-order decoupling sequences are known that achieve suppression up to arbitrary order q [i.e., leaving an $O(T^{q+1})$ unsuppressed error term in U(T)] [74–78], but for simplicity, in this work we restrict our attention to first-order sequences. Note that a common misconception is that DD is not effective against purely Markovian noise [79]; however this is not the case, essentially since even in the Markovian limit the bath can have a nonzero correlation time [80–83].

In general, decoupling using subgroups of \mathcal{P}_n will eliminate parts of H_{SB} , presenting an opportunity to selectively combine DD with QEC [57, 84] while taking advantage of biased noise [85, 86]. Recall that an [[n, k, d]]stabilizer code C is defined as the +1 eigenspace of a stabilizer group S of order 2^{n-k} (with commuting generators $\{S_j\}_{j=1}^{n-k}$). A stabilizer group is any subgroup of \mathcal{P}_n that excludes -I (which implies that it is Abelian). This factors the n-qubit Hilbert space into k logical qubits with an associated group of canonical logical operators $\mathcal{L} = \langle i, \overline{X}_j, \overline{Z}_j \rangle_{j=1}^k \subset \mathcal{P}_n$ (we use $\langle \cdots \rangle$ to denote a generating set; $|\mathcal{L}| = 4^{k+1}$) and n - k syndrome qubits (which can be used to detect errors). Specifically, syndrome qubits can be used to detect Pauli group terms that anticommute with at least one of the n-k stabilizer generators [3]. Since S is commutative, operators in S can be simultaneously diagonalized. This simultaneously neous diagonalization partitions the Hilbert space into an orthogonal sum of 2^{n-k} subspaces, each of dimension 2^k , corresponding to 2^{n-k} choices of eigenvalues of S_i for $j=1,\ldots,n-k$ (each such eigenvalue is ± 1), i.e., different values of the syndrome. We call these "syndrome spaces". \mathcal{C} is associated with the trivial (no error, corresponding to +1 eigenvalue of all S_i) syndrome and that two logical operators act identically on \mathcal{C} if they only differ by a stabilizer element, i.e., the full group of logical operators is C(S) = SL, the centralizer of S in P_n .

For a code $\mathcal{C}\subset (\mathbb{C}^2)^{\otimes n}$, the stabilizer group \mathcal{S} is uniquely defined as

$$S = \{ P \in \mathcal{P}_n : \forall \, |\psi\rangle \in \mathcal{C}, \, P \, |\psi\rangle = |\psi\rangle \}. \tag{4}$$

Its centralizer is also uniquely defined. However, the group of canonical logical operators is not unique: one can choose arbitrary stabilizers $S_j^X, S_j^Z \in \mathcal{S}$ (j = 1, ..., k) and multiply the generators \bar{X}_j, \bar{Z}_j by those to obtain other choices of the group \mathcal{L} .

It is always possible to write the system-bath interaction [Eq. (1)] as

$$H_{SB} = H_{SB}^{\mathcal{S}} + H_{SB}^{\mathcal{L}} + H_{SB}^{\mathcal{E}}, \tag{5}$$

grouping together terms with similar action on the code space \mathcal{C} . Altogether, this expression has 4^n linearly independent terms corresponding to the elements of $\tilde{\mathcal{P}}_n$. Specifically, $H_{SB}^{\mathcal{S}}$ collects all the distinct 2^{n-k} Pauli operators $P_i \in \mathcal{S}$ (including $P_0 = I$); such terms act as the identity operator on the code space and, therefore, are harmless. $H_{SB}^{\mathcal{L}}$ collects all $2^{n-k}(4^k-1)$ Pauli operators in $(\mathcal{SL} \setminus \mathcal{S})/\{\pm 1, \pm i\}$; these terms leave the code space (and each of the other syndrome spaces) invariant but cause harmful logical errors, since by construction, the code cannot detect such errors. Finally, $H_{SB}^{\mathcal{E}}$ collects all remaining $2^{n+k}(2^{n-k}-1)$ terms in $\tilde{\mathcal{P}}_n \setminus (\mathcal{SL}/\{\pm 1, \pm i\})$, which correspond to potentially detectable errors: given the value of the syndrome one can either detect all of them or correct some of them.

A key observation is that the decoupling group is arbitrary and may, in particular, be formed from the stabilizer group S and/or the group of canonical logical operators \mathcal{L} of a stabilizer code \mathcal{C} , or even a subset of either one [58, 72, 87–90]. One approach for selectively combining DD with QEC, which leverages this flexibility, is to use DD only to suppress the errors between or within the 2^{n-k} syndrome spaces, including the code space C(i.e., errors in $(\mathcal{SL} \setminus \mathcal{S})/\{\pm 1, \pm i\}$). Errors causing transitions between the syndrome spaces (i.e., elements of $\mathcal{P}_n \setminus (\mathcal{SL}/\{\pm 1, \pm i\})$ are the ones the code aims to detect or correct. The former are decoupled by $\mathcal{G} = \mathcal{L}$ while the latter are decoupled by $\mathcal{G} = \mathcal{S}$. For any choice of \mathcal{G} , the decoupling condition is that there exists at least one element of \mathcal{G} that anticommutes with a given error. Based on these properties, it was shown in Ref. [58] that combining the full set of n + k stabilizer and canonical logical operator generators to generate the DD group $\mathcal{G} = \mathcal{S} \times \mathcal{L}$, optimally achieves the goal of decoupling the errors between or within the 2^{n-k} syndrome spaces, in the sense of minimizing the length of the DD pulse sequence. This is known as 'stabilizer-logical DD' (SLDD) and incurs a single sequence cost of $T=2^{n+k}\tau$, instead of $4^n\tau$ in the case of full Pauli group decoupling. The savings are substantial when $k \ll n$. Adding QEC syndrome measurements and recovery operations based on syndrome measurements then detects the remaining errors between syndrome spaces that DD did not suppress [i.e., the $O(T^{q+1})$ terms in the case of a q'th-order DD sequence or successfully corrects some of them, resulting in a hybrid QEC-DD strategy.

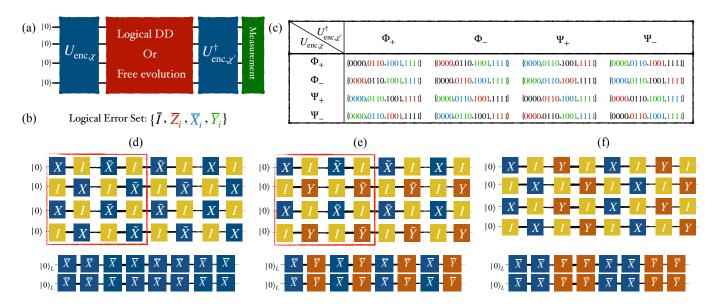


Figure 1. (a) Schematic of the experimental design: we encode a logical Bell state χ using the [[4,2,2]] code and let it undergo either free evolution or LDD for a duration τ . We then run the unencoding circuit for χ' and measure the resulting bitstring. The encoding and unencoding circuits together take up to 5μ s. (b) The logical error set that relates the four Bell states as a result of a logical operator acting on one (i=1 or 2) of the two logical qubits. (c) Interpretation of different bitstrings for an encoded state $(U_{\text{enc},\chi})$, depending on the unencoding $(U_{\text{enc},\chi'}^{\dagger})$ circuit used. The color code indicates that the encoded state has undergone either no error (black) or one of the logical errors (red, blue, green) indicated in (b). The DD sequences used in our experiments are (d) Robust Logical XX (RLXX), (e) Robust Logical XY4 (RLXY4), and (f) physically Staggered XY4 (SXY4). Each panel shows both the physical implementation (top) and the logical interpretation of each sequence. A tilde in (d) and (e) denotes a negative π rotation, i.e., $X = R_X(\pi) = \exp(-i\pi\sigma^x)$ $\tilde{X} = R_X(-\pi) = \exp(i\pi\sigma^x)$, and likewise for Y.

II. THE QEC-LDD THEOREM

The main weakness of SLDD is that it still incurs a cost that is exponential in the number of physical qubits n. Here, we present a different hybrid approach, shifting more of the burden for handling correctable errors to the QEC code and using only the logical operators as DD pulses g_j . We call this approach 'logical DD' (LDD). Compared to SLDD, this approach has the desired effect of reducing the single sequence time-cost to $T=4^k\tau$ for first-order decoupling, independent of n. Table I summarizes the relative resource cost of full Pauli group DD, SLDD, and LDD.

A. Theorem statement and proof

The LDD sequence is purely responsible for decoupling the logical errors acting within each syndrome space, but unlike SLDD, it no longer decouples some of the physical errors between the syndrome spaces. Correcting these errors is accomplished by syndrome measurements in conjunction with error recovery.

For each code \mathcal{C} there is a choice of which errors to correct: for each of the $2^{n-k}-1$ nontrivial syndromes, one may choose a recovery operation; this choice determines which errors are successfully corrected and which result in a logical error. Once such a choice is made, we call the

Comparative DD resource cost					
Decoupling group \mathcal{G}	# of pulses	suppresses all errors afflicting code space / requires QEC			
Pauli group $\tilde{\mathcal{P}}_n$	2^{2n}	Yes / No			
SLDD	2^{n+k}	Yes / No			
LDD	2^{2k}	No / Yes			

Table I. The comparative resource cost for various DD methods in terms of the number of pulses required to suppress the most general system-bath Hamiltonian H_{SB} [Eq. (5)] acting on n qubits, and whether active error correction is required in addition. Full Pauli group DD and SLDD suppress all error mechanisms which can decohere code states, while LDD does not, as it leaves dealing with these error mechanisms for the QEC cycle. Consequently, LDD does not scale with n but with k, whereas Pauli group DD and SLDD both scale with n.

errors that are successfully corrected "correctable". We summarize this in the following theorem:

Theorem 1 (QEC-LDD). Consider the most general system-bath Hamiltonian acting on n qubits [Eq. (5)]. Let \mathcal{C} be an [[n,k,d]] stabilizer code (with any $d \geq 1$). Let $\mathcal{L} = \langle i,\overline{X}_j,\overline{Z}_j \rangle_{j=1}^k$ denote its group of canonical logical operators. Choose the decoupling group $\tilde{\mathcal{G}} = \mathcal{L}/\{\pm 1, \pm i\}$.

henceforth the LDD group. Then $\tilde{\mathcal{G}}$ decouples all logical errors, that is, every term in $H_{SB}^{\mathcal{L}}$, as well as some errors in $H_{SB}^{\mathcal{E}}$. There is a choice of recovery operation such that all remaining non-decoupled errors in $H_{SB}^{\mathcal{E}}$ are correctable by \mathcal{C} . Conversely, for any recovery operation chosen in this way, only those errors that are not decoupled by $\tilde{\mathcal{G}}$ are correctable.

Proof. We first establish when an error $E \in \mathcal{P}_n$ is decoupled. Let $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$ be the Hilbert space of n qubits. Then $\mathcal{C} \subset \mathcal{H}$ and all elements of \mathcal{P}_n (and, hence, of $\mathcal{L} \subset \mathcal{P}_n$ and $\mathcal{S} \subset \mathcal{P}_n$) are unitary operators on \mathcal{H} . We let $\mathcal{G} = \mathcal{L} = \{\pm g_j, \pm i g_j\}_{j=0}^{4^k-1}$. Consider a term in H_{SB} corresponding to some $E \in \mathcal{P}_n$. It is decoupled if and only if there exists $g \in \mathcal{G}$ anticommuting with E. Indeed, then $g^{\dagger}Eg = -E$ and

$$\langle E \rangle_{\mathcal{G}} = \frac{1}{|\mathcal{G}|} \sum_{h \in \mathcal{G}} h^{\dagger} E h = \frac{1}{|\mathcal{G}|} \sum_{h \in \mathcal{G}} (gh)^{\dagger} E g h = -\langle E \rangle_{\mathcal{G}},$$
 (6)

so $\langle E \rangle_{\mathcal{G}} = 0$. If there is no such g then all elements of \mathcal{G} commute with E (because every two elements of \mathcal{P}_n either commute or anticommute). Then all $g^{\dagger}Eg = E$ for all $g \in \mathcal{G}$, hence $\langle E \rangle_{\mathcal{G}} = E$ (i.e., the term is undecoupled). It follows that $\tilde{\mathcal{G}}$ decouples all logical errors, since every such error anticommutes with an element of \mathcal{G} . The errors in $H_{SB}^{\mathcal{E}}$ that anticommute with an element of \mathcal{G} are also decoupled. All terms in $H_{SB}^{\mathcal{S}}$ are undecoupled since they all commute with \mathcal{G} .

The key observation in the proof is that the statement of the theorem is invariant under Clifford maps, which we now show.

Now consider any Clifford unitary $V\colon\mathcal{H}\to\mathcal{H}$. Let $\mathcal{C}'=V\mathcal{C},\ d'\geq 1$ be the distance of the code $\mathcal{C}',\ \mathcal{G}'=\mathcal{L}'=V\mathcal{L}V^\dagger,\ \tilde{\mathcal{G}}'=\mathcal{G}'/\{\pm 1,\pm i\},\ H'_{SB}=VH_{SB}V^\dagger,$ etc. For a recovery map R_s (for every value s of the syndrome) for \mathcal{C} we can construct the corresponding recovery map $R'_s=VR_sV^\dagger$ for \mathcal{C}' . If the statement of the theorem is true for the original code \mathcal{C} , then it remains true for \mathcal{C}' because it only depended on the commutation relations between elements of \mathcal{P}_n and those are preserved under the group homomorphism $P\mapsto VPV^\dagger$. By applying this observation to V^\dagger instead of V we observe that the converse also holds.

In Lemma 1 in Appendix C, extending earlier methods for stabilizer code encoding [91–93], we show that for any code \mathcal{C} and any choice of the canonical logical operators \mathcal{L} there is a Clifford unitary V (which we may call "unencoding circuit") such that the corresponding code \mathcal{C}' consists of states of the form $|\psi\rangle \otimes |0\rangle^{n-k}$, where $|\psi\rangle \in (\mathbb{C}^2)^{\otimes k}$, and $\mathcal{L}' = V\mathcal{L}V^{\dagger} \simeq \mathcal{P}_k$ consists of Paulis $P \in \mathcal{P}_n$ acting trivially on the last n-k qubits. We call such code \mathcal{C}' a trivial code: its logical qubits can be chosen to coincide with the first k physical qubits and stabilizer generators can be chosen to be Z_{k+1}, \ldots, Z_n enforcing that the last n-k qubits are set to 0, and d'=1.

It follows that it is sufficient to prove the statement of the theorem for this trivial code. Hence, from now on we assume that C is the trivial code described above. $E \in \mathcal{P}_n$ is undecoupled if and only if it commutes with all Paulis acting on the first k qubits. This is true if and only if E acts trivially on the first k qubits. (Equivalently, $E \in \mathcal{P}_n$ is decoupled if and only if acts nontrivially on one or more of the first k qubits.) Measuring the stabilizers Z_{k+1}, \ldots, Z_n is equivalent to measuring the last n-k qubits. Any such measurement outcome $s = (s_1, \ldots, s_{n-k})$ corresponds to errors $E = SE_s$, where $E_s = \prod_{i=1}^{n-k} X_{k+i}^{s_j}$ and S is an arbitrary stabilizer. A (unitary) recovery operation R_s corrects such an error Eif and only if $R_s = S'E_s$. In particular, if we apply E_s corresponding to the measured syndrome s, we correct all non-decoupled errors, proving the first statement of the theorem. On the other hand, all such recovery maps act trivially on the first k qubits, so if E acts non-trivially on the first k qubits (i.e. if E is decoupled), it remains uncorrected by any such procedure.

We emphasize a few key points:

- The claim holds even for d = 2 (or d = 1), i.e., an error *detecting* code becomes an error *correcting* code subject to LDD.
- In fact, the notion of correctable errors used here ignores the code distance altogether. For any distance, the QEC-LDD protocol has at least 3k non-correctable single-qubit errors (these are decoupled by LDD). For example, in the standard distance-d surface code there is a weight 1 error, which can be "corrected" to become a weight 2d-1 error. That does not contradict the theorem, because that error is suppressed by LDD.
- For each syndrome, the corresponding recovery operation is unique up to multiplication by stabilizers.
- All decoupled errors are not correctable; it is the role of LDD to suppress such errors.

B. Illustration of the QEC-LDD Theorem: the [[4, 2, 2]] code

To illustrate the QEC-LDD construction, we analyze the example of the [4, 2, 2] code (see Appendix A for a

Consider the group \mathcal{L} and divide the physical qubits into three classes according to the action of the elements of \mathcal{L} on a particular qubit. 1. Every element of \mathcal{L} acts as I on that qubit. 2. Every element of \mathcal{L} acts as I or one other Pauli (X, Y, or Z) on that qubit. 3. For every Pauli in $\{I, X, Y, Z\}$ there is $P \in \mathcal{L}$ which acts like that Pauli on that qubit. Let l_1, l_2, l_3 be the number of physical qubits in each class. Restrict every element of \mathcal{L} to the qubits from class 3. Such a restriction does not change the commutation relations. Let the resulting group be \mathcal{L}' . We have $\mathcal{P}_k \simeq \mathcal{L} \simeq \mathcal{L}'$, which in turn is embedded in \mathcal{P}_{l_3} . Therefore, $l_3 \geq k$. The number of uncorrectable single-qubit errors is $2l_2 + 3l_3 \geq l_3 \geq 3k$. E.g., for the [[4, 2, 2]] code $l_2 = 0, l_3 = 3, k = 2, 2l_2 + 3l_3 = 9 > 6 = 3k$.

review of this code). The stabilizer group of this code is $S = \langle XXXX, ZZZZ \rangle$. A set of logical operators for the code can be chosen such that $\overline{X}_1 = XIIX$, $\overline{X}_2 = IIXX$, $\overline{Z}_1 = IIZZ$ and $\overline{Z}_2 = ZIIZ$, i.e., $\mathcal{L} = \langle i, XIIX, IIXX, IIZZ, ZIIZ \rangle$. Up to a global phase, $\{XIIX, IIXX, IIZZ, ZIIZ\}$ is the generator set of the 16-element decoupling group $\tilde{\mathcal{G}} = \{g_j\}$. This LDD group suppresses every logical error E in the $4 \times 16 = 64$ -element $\mathcal{SL}/\{\pm 1, \pm i\}$: the group-averaged logical system-bath Hamiltonian $\langle H_{SB}^{\mathcal{L}} \rangle_{\mathcal{G}}$ vanishes, as every one of its terms is a logical operator that anticommutes with an element of \mathcal{G} .

The errors that are not suppressed are easily seen to be the 16-element set $\mathcal{E} = \{P \otimes Q \otimes P \otimes P : P, Q \in \{I, X, Y, Z\}\}$, since they commute with all elements of \mathcal{G} . Each such error is detectable because it anticommutes with at least one element of \mathcal{S} . Moreover, every such error is correctable since it is equal to the product of the stabilizer $P \otimes P \otimes P \otimes P$ and $I \otimes R \otimes I \otimes I$, where $R = Q^{\dagger}P$ can be determined (up to the global phase) from the value of the syndrome. The recovery operation is then the weight-one operator $I \otimes R \otimes I \otimes I$.

Ignoring phases, the Pauli group on 4 physical qubits, $\tilde{\mathcal{P}}_4$, has $4^4 = 256$ elements. Of these, 64 are the elements of $\mathcal{SL}/\{\pm 1, \pm i\}$, 16 are the unsuppressed errors \mathcal{E} , which leaves 176 other elements $[P] = \{\pm P, \pm iP\} \in \tilde{\mathcal{P}}_4$ such that corresponding $P \in \mathcal{P}_4$ anticommutes with at least one element of \mathcal{G} and is, therefore, suppressed by it.

Appendix C illustrates the unencoding of the [[4, 2, 2]] code. In this case, \mathcal{P}_2 decouples all errors on the first two (data) qubits and only X_3 or X_4 are left as undecoupled bit-flip errors on the last two qubits.

C. Avoiding the use of virtual-Z gates

One of the limitations of our experimental setup is that physical Z gates are unavailable: fixed-frequency transmons utilize virtual-Z gates [94, 95]. Such gates are problematic for DD [95]. We now demonstrate that there are DD sequences satisfying the conditions of the QEC-LDD theorem which do not use any physical Z pulses. To construct such a sequence, we first choose generators of \mathcal{L} that do not involve physical Z gates. An example choice would be $\mathcal{L} = \langle i, XIIX, IIXX, IIYY, YIIY \rangle$. One could follow the construction below with this choice. However, in order to improve the ability of the DD sequence to suppress ZZ crosstalk we suggest a different choice. First, multiply the original generators of \mathcal{L} (that is, XIIX, IIXX, IIZZ, ZIIZ) by the stabilizers XXXX, XXXX, XXXX, and ZZZZ, respectively, to obtain $\mathcal{L}' = \langle i, IXXI, XXII, XXYY, IZZI \rangle$. Then, after some group operations, we can write the same group using a different set of generators containing no Z's: $\mathcal{L}' = \langle i, XXII, IIYY, IYIY, XIXI \rangle$. The DD sequence consisting of two repetitions of XIXI, IYIY, XIXI, IIYY, XIXI, IYIY, XIXI, XXII then corresponds to the decoupling group $\mathcal{L}'/\{\pm 1, \pm i\}$, and involves no Z

gates.

This sequence was obtained as follows: denote the generators of \mathcal{L} (or \mathcal{L}') as h_1, h_2, h_3, h_4 and choose any Gray code — a sequence of bitstrings a_0, \ldots, a_{15} such that neighboring bitstrings differ in only a single digit. Finally, pick $g_j = h^{a_j}$ (for $j = 0, \ldots, 15$), where a_j is interpreted as a multiindex (i.e. $g_j = \prod_{k=1}^4 h_k^{(a_j)_k}$). The pulses of the DD sequence are $g_j g_{j+1}$ for $j = 0, \ldots, 15$.

However, in our experiments we found that a decoupling group generated by a subgroup of $\mathcal L$ and a subgroup of $\mathcal S$ works better in practice because of its robustness and ability to suppress crosstalk. The next section is dedicated to demonstrating experimentally how the hybrid QEC-LDD protocol results in a significant fidelity enhancement of entangled logical qubits.

III. EXPERIMENTAL DESIGN AND METHODOLOGY

The data for the experiments we report here were collected on two separate occasions from a total of 24 sets of four-qubit experiments run on the ibm_kyiv [96] quantum processor. Dataset 1, using 14 sets of qubits, was collected during the week of August 12th (2024). Dataset 2, using 10 sets of qubits, was collected on March 1st and 2nd (2025). These 10 four-qubit sets differ from those in dataset 1 in order to test the robustness of our results. The figure captions below specify the dataset. For each data point in our results, fidelity is independently calculated for each qubit set and then bootstrapped by resampling. The mean fidelity and standard deviation are derived from the bootstrapped data, where larger error bars indicate greater variability among the qubit sets.

The ibm_kyiv processor consists of coupled, fixedfrequency transmons [97]. Such qubits exhibit an alwayson interaction between adjacent pairs [98]. This crosstalk gives rise to weight-2 error terms that correspond to logical errors with our [4, 2, 2] code choice. Other errors of weight ≥ 2 , if present, would likewise correspond to logical errors. To demonstrate that LDD can suppress all logical errors, we design an experiment that enables the unequivocal detection of such errors. To this end, we use the two logical qubits of the [4, 2, 2] code to create logical Bell states $|\Phi_{\pm}\rangle = (|\overline{00}\rangle \pm |\overline{11}\rangle)/\sqrt{2}$ and $|\Psi_{\pm}\rangle = (|\overline{01}\rangle \pm |\overline{10}\rangle)/\sqrt{2}$. The corresponding encoding circuits $U_{\text{enc},\chi}$, where $\chi \in \{|\Phi_{\pm}\rangle, |\Psi_{\pm}\rangle\}$, create two copies of physical Bell states, followed by the application of a logical controlled-NOT ($\overline{\text{CNOT}}$) to generate the logical Bell states (see Appendix A).

To estimate the fidelity of a logical Bell state χ , we start from the physical ground state $|0000\rangle$, encode into the state of interest by applying $U_{\text{enc},\chi}$, and then let it evolve for some time τ , either freely or subject to DD.

If after time τ we unencode by applying $U_{\text{enc},\chi}^{\dagger}$, then measuring the bitstring 0000 would indicate that no error occurred.

No DD	XY4	UR_6	UR_8	UR_{10}	UR_{18}	$RGA8_a$
$ \Phi_{+}\rangle \ 65.44 \pm 0.24$	83.94 ± 0.18	83.08 ± 0.19	83.11 ± 0.19	82.05 ± 0.19	81.43 ± 0.20	84.20 ± 0.18
$ \Phi\rangle~61.23\pm0.24$						
$ \Psi_{+}\rangle \ 66.18 \pm 0.23$	84.68 ± 0.18	84.17 ± 0.18	83.37 ± 0.19	82.46 ± 0.19	81.21 ± 0.19	84.65 ± 0.18
$ \Psi_{-}\rangle \ 65.43 \pm 0.23$	86.57 ± 0.17	85.20 ± 0.18	85.53 ± 0.18	84.17 ± 0.18	83.44 ± 0.17	86.38 ± 0.17

Table II. Fidelity (%) of the encoded logical Bell states without and with DD, for a variety of pulse sequences. Experiments were performed on fourteen different sets of qubits on ibm_kyiv , with 4000 shots per set. The fidelities were computed independently for each set and then bootstrapped. The reported values represent the bootstrapped averages with 1σ uncertainty.

If, instead, we obtain any of the odd-Hamming weight bitstrings, this would signal detection of a physical error, as such states are not in the codespace. The final possibility is that we obtain one of the other even-Hamming weight bitstrings $\{0110, 1001, 1111\}$, corresponding to a logical error. This procedure can be used to estimate the probability of finding the ground state as the empirical fraction of ground state measurement outcomes, which is also the fidelity of the logical Bell state χ .

To obtain the fidelity of a logical error, we first observe that each of the logical Bell states is related to the other three logical Bell states through a specific logical error: $\chi' = \overline{O}\chi$, where $\overline{O} \in \{I, \overline{X}_i, \overline{Y}_i, \overline{Z}_i\}$ and i=1 or 2; i.e., a logical operator that acts on either of the two logical qubits. We thus proceed as follows: instead of applying $U_{\text{enc},\chi}^{\dagger}$, at $t=\tau$ we deliberately unencode into a different logical Bell state χ' [see Fig. 1(a)]. In this manner, each of the even-weight bitstrings gives us a measure of the occurrence of one of the logical errors $\mathcal{E}_{\overline{O},\tau}$, where \overline{O} is determined by the unencoding we choose [see Fig. 1(b,c)]. Using this methodology, we can detect the occurrence of different types of logical errors and quantify the associated error probability.

In more detail, each row in Fig. 1(c) shows a different initially encoded logical Bell state. Each column corresponds to one of the unencoding circuits. In each unencoding scenario, the different bitstrings indicate that the state being unencoded is either the initial logical Bell state or some other logical Bell state. For example, consider the case where we originally encode $|\Phi_+\rangle$ and then unencode into $|\Phi_-\rangle$ at $t=\tau$. If the state being unencoded is indeed $|\Phi_-\rangle = \overline{Z}_i \, |\Phi_+\rangle$, then the 0000 bitstring signals that a Z logical error (\overline{Z}_i) has occurred. However, if no error has occurred, then due to unencoding into $|\Phi_-\rangle$, the result should be the 0110 bitstring. Generalizing, it is possible to detect the different logical errors using the experiments indicated in Fig. 1(c).

However, we specifically choose the unencoding so that it is always the 0000 bitstring that corresponds to the occurrence of the logical error operator in which we are interested. We make this choice since $|0000\rangle$ is the ground state of the system and therefore is robust against relaxation errors. This strongly increases the likelihood that the detected errors are purely logical and are unaffected by thermal relaxation. Note that due to this choice, the bitstring that corresponds to the fidelity $\mathcal{F}_{\chi}(t)$ of the pre-

pared initial state varies [black color-coded bitstrings in Fig. 1(c)].

From here on, we use the notation $\chi \mapsto \chi'$ to denote the procedure of preparing the encoded logical Bell state χ and unencoding into χ' , i.e., of using the encoding unitary $U_{\text{enc},\chi'}$.

We note that logical state tomography is an alternative method for learning about the performance of LDD, and in particular, for certifying the entanglement of logical Bell states. However, the [[4,2,2]] code imposes some additional challenges in performing logical tomography, which precluded its use in our work; see Appendix D for details.

A. Physical dynamical decoupling improves logical Bell state fidelity

We first show that we can substantially improve the logical Bell fidelity by padding the idle gaps of the encoding circuits with physical (as opposed to logical) DD sequences. An *idle gap* is a temporal circuit segment during which no gates are applied. Such gaps occur, e.g., when a pair of qubits is involved in a two-qubit gate that takes much longer than a single-qubit gate simultaneously being applied to another qubit; the latter is then idle after the completion of the single-qubit gate, while awaiting the completion of the two-qubit gate.

As shown in Fig. 2, we inserted various DD sequences into the idle gaps of Bell state circuits. This includes XY4 [40], universally robust sequences UR_n [62] for n=6,8,10,18, and $\operatorname{RGA8}_a$ [61]; the former are illustrated along with robust variants in Fig. 1(d-f).

We performed the experiments (during the week of August $12^{\rm th}$ 2024) on fourteen different sets of qubits of ibm_kyiv and report the average fidelity values in Table II. In these experiments, we encoded and unencoded the same state, i.e., used $U_{\rm enc,\chi}$ along with $U_{\rm enc,\chi}^{\dagger}$. We see that using DD, the encoding fidelity improves by $\sim 20\%$ for all four logical Bell states, with XY4 and RGA8_a being the top performers (we attribute the lower performance of the longer UR_n sequences to pulse interference effects [95]).

In all, these results demonstrate that logical Bell state preservation can benefit significantly from physical DD.

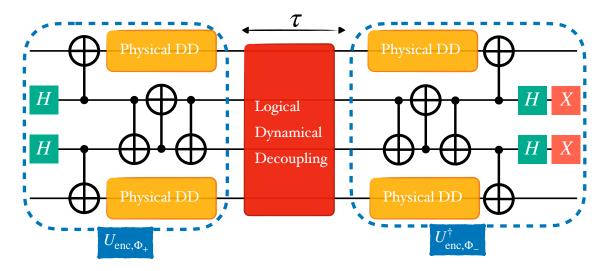


Figure 2. Circuit schematic for encoding $|\Phi_{+}\rangle$ and unencoding $|\Phi_{-}\rangle$ with physical and logical DD. Physical XY4 dynamical decoupling sequences (yellow boxes) are inserted into the idle gaps of the circuit. The code subspace is protected with LDD sequences. This is the DD-protected version of the circuit shown in Fig. 1(a).

Having established the utility of physical DD, we proceed to combine it with LDD in the next section.

B. Logical error suppression and detection by LDD

For all our experimental results, 'No DD' refers to [[4,2,2]] encoding without any physical or logical DD. In all other experiments, we use physical XY4 to pad all the encoding and unencoding circuits gaps, and combine them with various flavors of LDD. This choice allows us to clearly assess the improvements introduced by DD, compared to relying solely on the code's error detection capabilities.

1. Logical Z errors

We start by gradually increasing the time delay τ (up to 55 μ s) between the encoding and unencoding without DD. This situation is relevant in the context of QEC experiments. For example, one could prepare an encoded qubit and then leave it idle while other logical operations are applied to other encoded qubits [19].

We perform the \overline{Z}_i error detection using the logical Bell states $|\Psi_+\rangle$ and $|\Phi_+\rangle$ as discussed in Section III, and display the results in Fig. 3. As can be seen in Fig. 3(a) and (c), without LDD the free evolution fidelity (denoted as No DD) decays rapidly and exhibits ZZ crosstalk-induced oscillations. Fig. 3(b) and (d) show that logical Z errors accumulate over time. We next use two types of LDD sequences to suppress this effect: Logical XX (LXX) and Logical XY4 (LXY4). Physicallevel schematics of these LDD sequences are shown in the boxed four-pulse sequences of Fig. 1(d) and (e), respectively (disregarding the tilde notation). Their logical-

level counterparts are shown at the bottom of Fig. 1. We generate both LXX and LXY4 using the native logical operations of the [[4,2,2]] code. For example, in the upper part of Fig. 1(d), reading the first column from top to bottom yields $XIXI = \overline{XX}$, which is the \overline{XX} column at the bottom of Fig. 1(d). Reading the pulse sequences from left to right, the staggering (appearance of delays as indicated by the identity operations) is deliberately introduced to suppress crosstalk at the physical level [65]; see Appendix B for more details.

The corresponding results are shown as dashed lines in Fig. 3. At short times ($\leq 15\mu s$), these results are better than those without DD because some errors (including ZZ crosstalk) are suppressed by both LXX and LXY4. However, at longer times, the benefit is lost and, moreover, small oscillations appear that indicate the presence of coherent errors [99]. To overcome this, we create robust versions of LXY4 and LXX by ensuring that all physical qubits in the code undergo physical DD sequences robust to pulse errors. Specifically, we use the universally robust (UR) sequence family [62], and ensure that each physical qubit undergoes a UR₄ sequence, i.e., XXXX or YYYY, where a tilde denotes an X or Y rotation by $-\pi$ instead of π . These robust versions, which we call RLXX and RLXY4, are the full sequences shown in Fig. 1(d) and (e). The performance of these sequences is shown by the solid lines in Fig. 3 and exhibits a significant improvement. Notably, the logical Bell state fidelities decay more slowly and without oscillations (panels (a) and (c)), and the logical Z errors are strongly suppressed (panels (b) and (d)).

In addition, we apply the physical staggered XY4 (SXY4) sequence. This corresponds to applying a single XY4 sequence to each physical qubit but in a staggered manner to reduce crosstalk [65]. This sequence, shown in Fig. 1(f), also corresponds to an LDD sequence for the

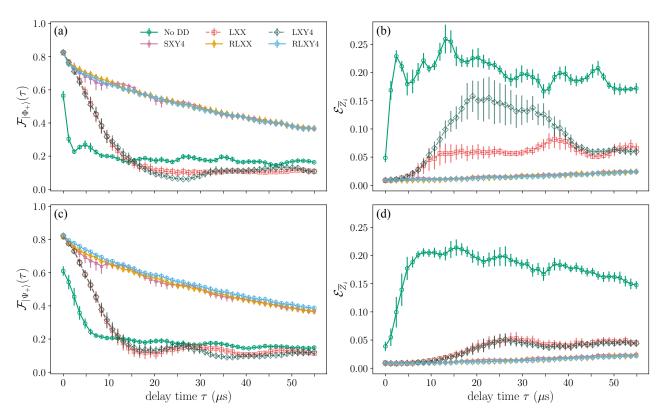


Figure 3. Performance of LDD (dataset 1). The experiments encode the logical Bell states $|\Phi_{+}\rangle$ in (a) and (b), and $|\Psi_{+}\rangle$ in (c) and (d). We then unencode in the logical Bell state related through \overline{Z}_{i} on either of the logical qubits; i.e., $|\Phi_{+}\rangle \mapsto |\Phi_{-}\rangle$ in (a) and (b), and $|\Psi_{+}\rangle \mapsto |\Psi_{-}\rangle$ in (c) and (d). In this setting, the frequency of 0110 outcomes [(a) and (c)] is a measure of the fidelity of the originally encoded state, while 0000 corresponds to the detection of a \overline{Z}_{i} error [(b) and (d)]. Without any DD, the fidelity is low (green in (a) and (c)) and the probability of logical errors is high [green in (b) and (d)]. Standard LDD sequences (LXX and LXY4; dashed light red and blue) improve the fidelity and logical error probability at short times ($\lesssim 15\mu$ s) but their performance declines at longer times: they exhibit coherent errors as seen in the oscillations of the corresponding curves. Robust sequences (RLXX and RLXY4; solid dark red and blue) are the top performers; their ability to suppress the \overline{Z}_{i} error is particularly noteworthy. Error bars are 1σ standard deviation after bootstrapping the data.

[[4,2,2]] code (bottom of panel (f)). However, it is not a universal logical sequence, but rather a purely logical-Z-error suppressing sequence. Interestingly, Fig. 3 shows that SXY4 performs on par with RLXX and RLXY4. This finding signals that the dominant logical errors are of Z-type. We confirm this in the following.

2. Logical X and Y errors

So far, we have only discussed the detection and suppression of Z-type logical errors. In order to detect X and Y-type logical errors, we proceed in analogy to the \overline{Z} error detection procedure, but with a different unencoding step. Namely, we start by encoding the logical $|\Phi_{+}\rangle$ state, and then unencode in either $|\Psi_{+}\rangle$ to detect X-type errors or in $|\Psi_{-}\rangle$ to detect Y-type errors. The results are shown in Fig. 4 where, for comparison, we have also included the no-DD $|\Phi_{+}\rangle \mapsto |\Phi_{-}\rangle$ and $|\Psi_{+}\rangle \mapsto |\Psi_{-}\rangle$ results shown in Fig. 3(b,d), which measure Z-type logical errors. It is clear from Fig. 4 that while unsuppressed logical errors Z accumulate rapidly, logical errors of X

and Y-type grow much more slowly. This confirms that ZZ crosstalk is the main source of logical errors.

3. LDD with postselection

Having implemented LDD, we can further improve the results by using the error detection capability of the [[4,2,2]] code, which allows us to perform postselection. We do so by discarding any measurement outcome outside the logical basis (which would be the result of a physical error); i.e., we only keep measurement results corresponding to the bitstrings $\{0000,0110,1001,1111\}$.

In Fig. 5(a) and (b) we show the fidelities of $|\Phi_+\rangle \mapsto |\Phi_-\rangle$ and $|\Psi_+\rangle \mapsto |\Psi_-\rangle$ for dataset 2 (Section III). Without DD, the fidelity decays rapidly while exhibiting crosstalk oscillations, as seen for dataset 1 in Fig. 3. The fidelity improves once we perform postselection as described above. This leaves us with bitstrings corresponding to logical states, but logical errors still reduce the fidelity. Using LDD in the form of the RLXY4 sequence – which suppresses both logical errors and phys-

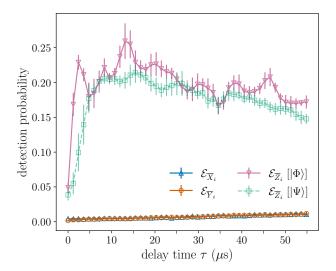


Figure 4. Detection probability (i.e., probability of the 0000 bitstring) of logical X, Y, and Z-type errors (dataset 1). The lower probability of logical X and Y-type errors signals that the main source of logical errors is ZZ crosstalk. These experiments are performed without DD.

ical errors – followed by postselection, we achieve fidelities > 82% for $|\Phi_{+}\rangle$ and > 87% for $|\Psi_{+}\rangle$ over a 55 μ s period. The average fidelities over the same period are 91.12% for $|\Phi_{+}\rangle$ and 93.66% for $|\Psi_{+}\rangle$. For the particular set of qubits numbered {58,59,60,61} (light dashed lines), we find that the combination of LDD and postselection yields average fidelities of \approx 95.44% for $|\Phi_{+}\rangle$ and \approx 94.78% for $|\Psi_{+}\rangle$, which is significantly higher than when we use only the error detection capability of the code: \approx 44.81% for $|\Phi_{+}\rangle$ and \approx 48.47% for $|\Psi_{+}\rangle$.

4. Beyond breakeven and the state-of-the-art

Fig. 5 also includes results for physical (unencoded) Bell states. Here we show only the best Bell pair among all pairs we tested, both without DD and with staggered (crosstalk-robust [65]) XY4. The former ('Best Physical No DD') exhibits strong crosstalk-induced oscillations with an overall fidelity comparable to that of the mean logical encoded Bell pairs case without DD (No DD+PS). Adding SXY4 significantly improves the fidelity and outperforms even the mean fidelity of logical Bell pairs with LDD (RLXY4). This shows that LDD by itself is not better than working with physical qubits and a crosstalkrobust DD sequence. However, physical Bell pairs with SXY4 are significantly worse than RLXY4+PS, i.e., the case of LDD with postselection on the results of the [4, 2, 2] code. This constitutes clear evidence of beyondbreakeven performance for our QEC-LDD strategy.

Overall, it is clear that the combination of LDD and postselection significantly boosts the fidelity of logical Bell states. Moreover, our results improve upon the current state-of-the-art using superconducting transmon

qubits. E.g., Ref. [100] used distance d=2 surface codes to encode the four logical Bell states, with a peak encoding fidelity of 79.5%. In contrast, we find an average postselected encoding fidelity of 98.05%. The averaging is over the 10 sets of qubits and over the two logical Bell states we prepare.

Ref. [101] used the heavy-hex surface code with variable distance d and reported a peak postselected fidelity of 93.7% after the first stabilizer round for d=2, declining to ≈ 0.3 after five rounds, which corresponds to $\approx 27\mu s$ on ibm_torino.

In contrast, we find an average peak postselected encoding fidelity of 98.05% that declines to 84.87% after $55\mu s$, and a peak postselected fidelity of 98.00% (also averaged over the two logical Bell states we prepare) for the best set of qubits, that declines to 92.89% after $55\mu s$.

C. Physical error suppression and detection by LDD

The LDD sequences suppress not only logical errors but physical errors as well. To see this, consider, e.g., the LXX sequence [Fig. 1(d)]. As described in Section III B 1, reading the pulse sequence vertically, each time step of this sequence operates in the logical subspace as \overline{XX} . Simultaneously, reading the sequence horizontally over a complete round of LDD, all four physical qubits undergo the physical XX sequence (i.e., $X - \tau - X - \tau$), which suppresses the set of physical errors $\{Y_i, Z_i\}_{i=1}^4$. Similarly, for the RLXY4 sequence [Fig. 1(e)], each physical qubit undergoes UR₄. which robustly [62] suppresses the sets of physical errors $\{Y_i, Z_i\}_{i=1,3}$ (due to the $X\tilde{X}\tilde{X}X$ sequence) and $\{X_i, Z_i\}_{i=2,4}$ (due to the $Y\tilde{Y}\tilde{Y}Y$ sequence).

One way to gauge the impact of this suppression of physical errors is shown in Fig. 6, which displays the percentage of discarded data per circuit as a result of postselection over $55\mu s$ (out of 4000 shots), comparing No DD to RLXY4 for the two logical Bell states $|\Phi_+\rangle$ and $|\Psi_+\rangle$. The results are obtained through bootstrapping across the ten sets of four qubits used in these experiments. Significantly less data are discarded with RLXY4 than without DD. Since discarded data correspond to the detection of physical errors, this means that LDD not only enhances the fidelity of logical Bell states by suppressing logical errors, but also reduces the occurrence of physical errors.

We can go further and use Algorithmic Error Tomography [54] to identify specific physical error types. For example, the 0100 and 0010 bitstrings correspond, respectively, to physical Z_2, Z_4 and Z_1, Z_3 , errors in the $|\Phi_+\rangle \mapsto |\Phi_-\rangle$ logical Bell state experiment. In Fig. 7 we show the corresponding relative bitstring counts, which are measures of detecting either of these errors. Additionally, we show the 0000 and 0110 relative bitstring counts, which correspond to the logical fidelity and the logical Z error, respectively. Without LDD (dashed), we observe that the logical fidelity is low and both logical and phys-

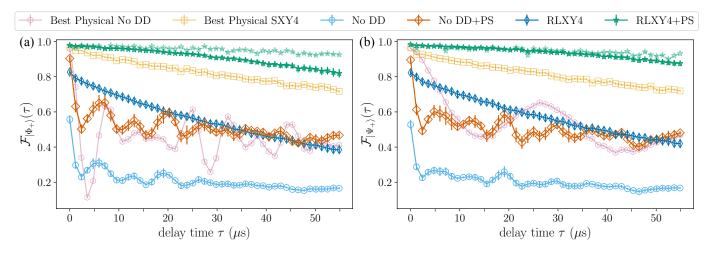


Figure 5. Fidelity of logical Bell states (dataset 2) for (a) $|\Phi_{+}\rangle$ (using $|\Phi_{+}\rangle \mapsto |\Phi_{-}\rangle$) and (b) $|\Psi_{+}\rangle$ (using $|\Psi_{+}\rangle \mapsto |\Psi_{-}\rangle$). In both cases, without DD the fidelity drops to 20% in $\approx 10\mu s$ (No DD). (For ibm_kyiv, median $T_1=279.92~\mu s$, median $T_2=111.926~\mu s$.) Using error detection and applying postselection, the fidelity improves to $\approx 35\%$ (No DD+PS). Using LDD alone is roughly equivalent to No DD+PS, but without the crosstalk oscillations (RLXY4). When we combine LDD with postselection, the fidelity increases substantially and remains above 82% [in (a)] and 87% [in (b)] over a 55 μ s period (RLXY4+PS). The green dashed (star) lines in each panel represent the results obtained from running the respective protocol on the best-performing set of physical qubits (using qubits 58, 59, 60, 61). In this case RLXY4+PS yields $\mathcal{F}\approx 92\%$ after 55 μ s (for both cases), significantly higher than the fidelity using postselection after error detection alone. Also shown is the best physical Bell state prepared on the same set of qubits (yellow squares), which, without DD (Best Physical No DD) performs comparably to No DD+PS, though with much larger crosstalk-induced fidelity oscillations. Applying staggered XY4 (Best Physical SXY4) removes the crosstalk and improves the best physical Bell state fidelity beyond RLXY4's (decaying to $\mathcal{F}\approx 71\%$ after 55 μ s), but remaining well below the performance of RLXY4+PS.

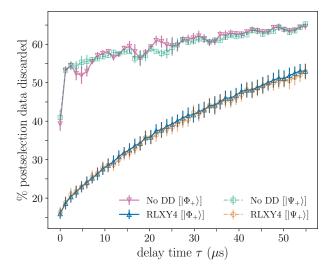


Figure 6. The percentage of data discarded per circuit as a function of the delay time between encoding and unencoding (dataset 2). While LDD suppresses logical errors, it also reduces certain physical errors, as evidenced by the lower data discard rate when using LDD (i.e., fewer physical errors are detected).

ical errors increase and oscillate. With DD (solid), the RLXY4 sequence strongly suppresses both types of errors.

These results demonstrate that, as claimed, LDD se-

quences suppress both physical and logical errors.

IV. DISCUSSION

The operation of QEC codes is adversely affected by the occurrence of logical errors that the code cannot detect or correct. Here, we have shown how to combine QEC with dynamical decoupling implemented in terms of the logical operators of the code, resulting in a hybrid QEC-LDD strategy that is significantly more effective than either QEC or LDD alone.

We designed our LDD sequences to simultaneously perform logical error suppression and to be robust DD sequences at the physical level, resistant to both control errors and crosstalk. Our results, using the [[4,2,2]] code and IBM transmon qubits, demonstrate a beyond-breakeven fidelity of entangled logical qubits.

Our findings address a need along the path toward fault-tolerant quantum computation: keeping codes relatively small and nimble while still effectively handling logical errors.

Future research should aim to optimize LDD sequences tailored to specific codes and integrate QEC-LDD into quantum algorithms. Another interesting future direction is optimization of QEC-LDD for tunable-coupler transmon devices; we present preliminary results in Appendix ${\bf E}$.

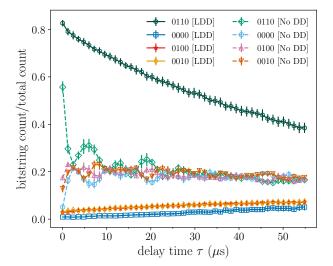


Figure 7. The relative count of the $\{0000,0110,0100,0010\}$ bitstrings in the $|\Phi_{+}\rangle \mapsto |\Phi_{-}\rangle$ experiments (dataset 2). The two even-weight bitstrings correspond to the logical fidelity (0000) and logical Z error (0110), and the two odd-weight bitstrings correspond to the detection of physical $\{Z_i\}_{i=1}^4$ errors. Dashed (solid) lines correspond to no DD (RLXY4). The LDD sequence, in addition to suppressing the logical errors and improving the logical fidelity, also strongly suppresses the occurrence of these physical errors.

ACKNOWLEDGMENTS

This material is based upon work supported by, or in part by, the Intelligence Advanced Research Projects Activity (IARPA), under the Entangled Logical Qubits program through Cooperative Agreement Number W911NF23-2-0216, by the U.S. Army Research Laboratory and the U.S. Army Research Office under contract/grant number W911NF2310255, and by the Defense Advanced Research Projects Agency under Agreement HR00112230006. The views, opinions and/or findings expressed are those of the author(s) and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government. This research was conducted using IBM Quantum Systems provided through University of Southern California's IBM Quantum Innovation Center. The views expressed are those of the authors and do not reflect the official policy or position of IBM or the IBM Quantum team.

Appendix A: [[4,2,2]] code

The [[4,2,2]] code is an error detection code that encodes k=2 logical qubits into n=4 physical qubits [102]. The stabilizer group is $\mathcal{S}=\{I,XXXX,YYYY,ZZZZ\}$. Defining the logical states

as

$$\begin{split} |\overline{00}\rangle &= \frac{|0000\rangle + |1111\rangle}{\sqrt{2}}, \quad |\overline{10}\rangle &= \frac{|0110\rangle + |1001\rangle}{\sqrt{2}}, \\ |\overline{01}\rangle &= \frac{|0011\rangle + |1100\rangle}{\sqrt{2}}, \quad |\overline{11}\rangle &= \frac{|1010\rangle + |0101\rangle}{\sqrt{2}}, \end{split} \tag{A1}$$

a set of logical operators for the code can be defined such that $\overline{X}_1 = XIIX$, and $\overline{X}_2 = IIXX$, up to multiplication by a stabilizer element. Therefore, we have $\overline{XX} = XIXI$. Similar definitions apply to the logical Z operators: $\overline{Z}_1 = IIZZ$ and $\overline{Z}_2 = ZIIZ$, allowing us to form the full logical Pauli group. Using the same definitions, we have $\overline{\text{CNOT}}_{12} = \text{SWAP}_{12}$ and similarly, $\overline{\text{CNOT}}_{21} = \text{SWAP}_{23}$, i.e., we can perform logical CNOTs by swapping the physical qubits.

Our particular choice of logical operators is motivated by the fact that \overline{XX} and \overline{YY} (i.e., the logical operators we use to implement the LDD sequences) have a natural staggering of their physical X and Y gates. In other words, in the implementation of the logical operators comprising LDD, nearest-neighbor qubits are always interleaved with an identity operation (e.g., IXIX as opposed to IXXI). This is critical for nearest-neighbor crosstalk cancellation [65].

To encode the logical Bell states $|\Psi_{\pm}\rangle$ and $|\Phi_{\pm}\rangle$ using the two logical qubits of the [[4, 2, 2]] code, consider:

$$\begin{split} |\Phi_{\pm}\rangle &= \frac{1}{\sqrt{2}} \big(|\overline{00}\rangle \pm |\overline{11}\rangle \big) & (A2a) \\ &= \frac{1}{\sqrt{2}} \big(|0000\rangle + |1111\rangle \pm |0101\rangle \pm |1010\rangle \big) & (A2b) \\ &= \overline{CNOT}_{21} \frac{1}{\sqrt{2}} \big[|0000\rangle + |1111\rangle \pm |0011\rangle \pm |1100\rangle \big] \\ &\qquad (A2c) \\ &= \overline{CNOT}_{21} \frac{1}{\sqrt{2}} \big[(|00\rangle \pm |11\rangle) (|00\rangle \pm |11\rangle) \big]. & (A2d) \end{split}$$

Thus, we proceed by preparing two *physical* copies of the target Bell state on the four physical qubits, then apply $\overline{\text{CNOT}}_{21}$, which creates the intended Bell state. The $|\Psi_{\pm}\rangle$ Bell states are prepared similarly.

Appendix B: Creating robust logical sequences

As discussed above, we use the logical operators $\overline{XX} = XIXI$ and $\overline{YY} = IYIY \equiv YIYI$ (along with identity operators) to generate the LDD groups $\mathcal{G}_{\text{LXY4}} = \{I, \overline{XX}, \overline{YY}, \overline{ZZ}\}$, and $\mathcal{G}_{\text{LXX}} = \{I, \overline{XX}\}$. Cycling over the group elements creates the four-pulse sequences LXY4 and LXX shown in the red boxes of Fig. 1(d,e).

Since the main source of logical errors is ZZ crosstalk, we optimize LDD so that it cancels such errors. Thus, in the case of LXX, we insert a stabilizer every other pulse-step to consistently apply a staggered sequence to all qubits. These sequences correspond to applying logical X and Y operators to both logical qubits.

To create the robust sequences, we mirror the original sequences (hence the eight-pulse sequences) and instead of X and Y, we apply $\tilde{X} = R_X(-\pi)$ $\tilde{Y} = R_Y(-\pi)$, respectively, such that each sequence undergoes a robust pulse sequence at the physical level. We also apply staggered physical XY4 to each qubit as shown in Fig. 1(f). This sequence is also inherently robust at the physical level since each qubit receives an XY4 sequence, which is robust to pulse errors [62]. However, it is not a universal decoupling sequence at the logical level, as seen in the figure (i.e., it lacks the ability to decouple arbitrary single logical-qubit errors).

Appendix C: The encoding/unencoding Lemma

Encoding of stabilizer codes is a familiar problem [91]. Here we present an explicit version of encoding circuits that appears to be simpler than other versions found in the literature [92, 93].

1. Lemma and proof

Lemma 1. For any [[n,k,d]] stabilizer code it is always possible to construct a unencoding circuit V^{\dagger} such that V^{\dagger} converts any group of canonical logical operators and a corresponding stabilizer group into single-qubit Pauli X and Z operators acting just on the first k physical data qubits and into single-qubit Pauli Z stabilizer operators on the remaining n-k ancilla qubits.

Proof. Our proof is constructive and relies on the basic Clifford group identities $\text{CNOT}_{i,j}(X_iX_j)\text{CNOT}_{i,j} = X_i$ and $\text{CNOT}_{i,j}(Z_iZ_j)\text{CNOT}_{i,j} = Z_j$, where $\text{CNOT}_{i,j}$ denotes the controlled-NOT gate with qubit i as the control and qubit j as the target. Note that this identity allows us to reduce the number of X operators.

Consider first the case k > 0.

- 1. Order the canonical logical operators and stabilizers as $\{\overline{X}_1, \overline{Z}_1, \dots, \overline{X}_k, \overline{Z}_k, S_1, \dots, S_{n-k}\}.$
- 2. Ensure that \overline{X}_1 consists only of X and I physical operators: apply $H = (X+Z)/\sqrt{2}$ and $(X+Y)/\sqrt{2}$ to the qubits where Z and Y, respectively, are present in the expression for \overline{X}_1 .
- 3. Apply SWAP_{1,j} if necessary to ensure X_1 is present in the expression for \overline{X}_1 .
- 4. Apply Z_1 if necessary to cancel the sign if the sign was -1.
- 5. Apply CNOT_{1,j} to eliminate all X components of \overline{X}_1 except X_1 .
- 6. Apply analogs of steps 2, 4, and 5 for \overline{Z}_1 .

7. Now note that all the other operators contain I_1 (indeed, we have $\overline{X}_1 = X_1$, $\overline{Z}_1 = Z_1$, and all other operators commute with these two, hence they cannot contain X_1 , Z_1 , or Y_1), so the task is reduced to the case with k := k - 1 and n := n - 1; proceed recursively.

Next, consider the case k=0. We have stabilizer operators $S_1, ..., S_n$ on n physical qubits and wish to transform them to $Z_1, ..., Z_n$.

- 1. Apply analogs of previous steps 2-5 to S_1 to transform it into Z_1 .
- 2. Recursively call this procedure to transform the remaining n-1 components of the remaining n-1 stabilizers to Z_2, \ldots, Z_n .
- 3. Apply CNOT_{1,j} gates to cancel the Z_1 components of the remaining n-1 stabilizers (if they had one). Note that they could not have had X_1 or Y_1 components because otherwise they would not commute with S_1 .
- 4. Apply X_j to cancel the sign of the remaining stabilizers if necessary.

2. Illustration using the [4, 2, 2] code

As an example, consider the [[4,2,2]] code. The stabilizer is $S = \langle X^{\otimes 4}, Z^{\otimes 4} \rangle$, and we choose the canonical logical operators as $\mathcal{L} = \langle \overline{X}_1 = X_1 X_4, \overline{X}_2 = X_1 X_2, \overline{Z}_1 = Z_1 Z_2, \overline{Z}_2 = Z_1 Z_4 \rangle$. The corresponding code basis states are

$$\begin{aligned} |\overline{00}\rangle &= \frac{1}{\sqrt{2}} |0000\rangle + |1111\rangle \\ |\overline{10}\rangle &= \frac{1}{\sqrt{2}} |0110\rangle + |1001\rangle \\ |\overline{01}\rangle &= \frac{1}{\sqrt{2}} |0011\rangle + |1100\rangle \\ |\overline{11}\rangle &= \frac{1}{\sqrt{2}} |1010\rangle + |0101\rangle \,. \end{aligned}$$
(C1)

Following the steps in the proof of Lemma 1, we obtain the following series of transformations:

$$\overline{X}_{1} = X_{1}X_{4} \stackrel{\text{CNOT}_{14}}{\longmapsto} X_{1} \longmapsto X_{1} \longmapsto X_{1} \longmapsto X_{1}$$

$$\overline{Z}_{1} = Z_{1}Z_{2} \longmapsto Z_{1}Z_{2} \stackrel{\text{CNOT}_{21}}{\longmapsto} Z_{1} \longmapsto Z_{1} \longmapsto Z_{1}$$

$$\overline{X}_{2} = X_{1}X_{2} \longmapsto X_{1}X_{2}X_{4} \longmapsto X_{2}X_{4} \stackrel{\text{CNOT}_{24}}{\longmapsto} X_{2} \longmapsto X_{2}$$

$$\overline{Z}_{2} = Z_{1}Z_{4} \longmapsto Z_{4} \longmapsto Z_{2}Z_{4} \stackrel{\text{CNOT}_{42}}{\longmapsto} Z_{2}$$

$$S_{1} = X^{\otimes 4} \longmapsto X_{1}X_{2}X_{3} \longmapsto X_{2}X_{3} \longmapsto X_{2}X_{3}X_{4} \longmapsto X_{3}X_{4}$$

$$\stackrel{H_{3}H_{4}}{\longmapsto} Z_{3}Z_{4} \stackrel{\text{CNOT}_{34}}{\longmapsto} Z_{3}$$

$$S_{2} = Z^{\otimes 4} \longmapsto Z_{2}Z_{3}Z_{4} \longmapsto Z_{2}Z_{3}Z_{4} \longmapsto Z_{3}Z_{4}$$

$$\longmapsto Z_{3}Z_{4} \longmapsto Z_{4} \longmapsto Z_{4}$$

$$(C2)$$

where we have only indicated the transformations once, for the operator that induces the transformation per the proof of Lemma 1; e.g., CNOT_{14} is induced by \overline{X}_1 but is applied to every operator in the first timestep, etc.

Thus, the unencoding circuit, i.e., the circuit mapping every logical and stabilizer operator from the [[4, 2, 2]] code to the corresponding operators of the trivial code, is

$$V^{\dagger} = \text{CNOT}_{34} H_4 H_3 \text{CNOT}_{24} \text{CNOT}_{42} \text{CNOT}_{21} \text{CNOT}_{14},$$
(C3)

where H_i is the Hadamard on the *i*th qubit.

After unencoding, we obtain the trivial code with $S_0 = \langle Z_3, Z_4 \rangle$ and $\mathcal{L}_0 = \langle X_1, X_2, Z_1, Z_2 \rangle$, that divides the code into k = 2 data qubits and n - k = 2 ancillas.

In the trivial code, error correction is as follows: apply IIXI when IIZI = -1 and IIIX when IIIZ = -1. Applying V^{\dagger} to these two recovery operators we find that to correct an undecoupled error in the original [[4, 2, 2]] code, one should apply IIIZ whenever XXXX = -1 and XXXI (or IIIX, since they are related by a stabilizer element) when ZZZZ = -1.

Appendix D: Issues involving the implementation of logical state tomography using the [[4,2,2]] code

A general two-qubit state requires information about 15 expectation values which requires 9 independent measurement settings (achieved by measuring each qubit in three complementary bases; e.g., X, Y, and Z) [103]. This holds for logical tomography of a logical two-qubit state as well. There are a few ways in which a logical observable can be measured. First, one can use stabilizer-measurement-like circuits where the logical operator is measured by executing a circuit composed of CNOT gates targeting an ancilla qubit that is measured to learn the logical measurement outcome. This method can further be paired with a round of syndrome extraction using additional ancilla qubits to learn whether the state was in the code space to begin with. This combination of measurements allows us to invoke the code's protection while simultaneously performing a measurement. The downside of this method is the additional overhead in CNOTs and ancilla qubits needed to perform the measurement protocol. In our case, this would require a substantial overhead in SWAP gates as well since the IBM QPU's heavy-hex lattice does not pair naturally with the [[4, 2, 2]] code. Using this logical measurement method would inevitably introduce more errors and reduce the accuracy of logical state tomography.

An alternative, less costly approach for performing logical measurements is to directly measure all data qubits of the code in lieu of introducing ancillas and additional CNOTs. However, this method is incompatible with ex-

tracting information about the entire stabilizer generator set while also performing the logical measurement. As a result, we would not be able to know with certainty that the system state was in the code space at the time of measurement. There is some nuance to this approach, which does allow us to learn information about stabilizers which commute qubit-wise (i.e., they share the same Pauli operator or I on the same qubit) with the logical operator being measured. For example, we can simultaneously measure $\overline{X}_1 = XIIX$, $\overline{X}_2 = IIXX$, $\overline{X}_1X_2 = XIXI$ and the stabilizer $S_1 = XXXX$ since these operators all qubit-wise commute. Performing the measurement of each data qubit in the X basis gives us information about each of these logical observables, as well as information about whether or not the system was in the logical code space with respect to the stabilizer generator S_1 . Unfortunately, this symmetry does not hold for the operators $\overline{Y}_1 = XIZY$, $\overline{Y}_2 = ZIXY$, and $S_2 = YYYY$ since these operators do not all commute qubit-wise. This means that we cannot simultaneously learn information about these logical observables and information about whether or not the system was in the code space with respect to the stabilizer S_2 in the same shot, reducing our total information about the system. This would lead to a less precise estimation of the logical density matrix, as we will count more experimental shots involving the system outside of the logical state space in our estimation of the logical expectation value.

For these reasons, we did not use logical state tomography in this work.

Appendix E: Logical dynamical decoupling with tunable couplers

Here we present additional experiments ibm_marrakesh, which features tunable couplers [104] unlike the always-on ZZ interaction in ibm_kyiv . Consequently, for this set of experiments, crosstalk is significantly reduced compared to the other set of results $(\lesssim 5 \text{ kHz vs tens of kHz})$. Nevertheless, a combination of dephasing and residual crosstalk accumulates, leading to logical errors. Figure 8 illustrates the performance of LDD sequences on this QPU, averaged over 9 different qubit sets. Fig. 8(a) [equivalent to Fig. 3(b)] shows that without DD, logical errors remain unsuppressed. However, both logical DD sequences we employ successfully suppress these errors. Fig. 8(b) [equivalent to Fig. 5(a)] demonstrates that using only postselection, the fidelity averages to $\mathcal{F} \approx 56.23\%$, whereas combining postselection with LDD increases it to $\mathcal{F} \approx 91.73\%$.

Optimizing LDD for specific types of tunable couplers to maximize the interplay between DD and couplers is an avenue for future work.

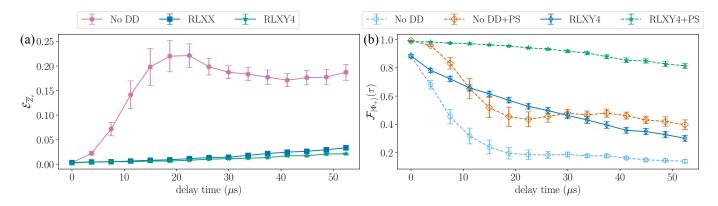


Figure 8. Performance of LDD on for an encoded $|\Phi_{+}\rangle$ Bell state on ibm_marrakesh for nine set of qubits. (a) shows the logical errors and (b) shows the postselection results. This device features tunable couplers that significantly reduce the crosstalk. Nevertheless, accumulation of dephasing errors over time (up to $\approx 55 \,\mu s$) lead to the appearance of logical errors. The same LDD sequence we used in the main text successfully suppresses these errors, as shown in (a). Additionally, LDD with postselection outperforms postselection alone, as shown in (b).

- [1] P. W. Shor, Scheme for reducing decoherence in quantum computer memory, Phys. Rev. A 52, R2493 (1995).
- [2] A. M. Steane, Error correcting codes in quantum theory, Phys. Rev. Lett. 77, 793 (1996).
- [3] D. Gottesman, Class of quantum error-correcting codes saturating the quantum hamming bound, Phys. Rev. A 54, 1862 (1996).
- [4] A. R. Calderbank and P. W. Shor, Good quantum error-correcting codes exist, Physical Review A 54, 1098 (1996).
- [5] D. P. DiVincenzo and P. W. Shor, Fault-tolerant error correction with efficient quantum codes, Physical Review Letters 77, 3260 (1996).
- [6] D. Aharonov and M. Ben-Or, Fault tolerant quantum computation with constant error, in *Proceedings of 29th Annual ACM Symposium on Theory of Computing (STOC)* (ACM, New York, NY, 1997) p. 176.
- [7] D. Gottesman, Theory of fault-tolerant quantum computation, Physical Review A 57, 127 (1998).
- [8] E. Knill, R. Laflamme, and W. H. Zurek, Resilient quantum computation, Science 279, 342 (1998).
- [9] D. Lidar and T. Brun, eds., Quantum Error Correction (Cambridge University Press, Cambridge, UK, 2013).
- [10] E. T. Campbell, B. M. Terhal, and C. Vuillot, Roads towards fault-tolerant universal quantum computation, Nature 549, 172 EP (2017).
- [11] D. G. Cory, M. D. Price, W. Maas, E. Knill, R. Laflamme, W. H. Zurek, T. F. Havel, and S. S. Somaroo, Experimental quantum error correction, Phys. Rev. Lett. 81, 2152 (1998).
- [12] J. Chiaverini, D. Leibfried, T. Schaetz, M. D. Barrett, R. B. Blakestad, J. Britton, W. M. Itano, J. D. Jost, E. Knill, C. Langer, R. Ozeri, and D. J. Wineland, Realization of quantum error correction, Nature 432, 602 (2004).
- [13] T. B. Pittman, B. C. Jacobs, and J. D. Franson, Demonstration of quantum error correction using linear optics, Physical Review A 71, 052332 (2005).
- [14] M. Takita, A. W. Cross, A. D. Córcoles, J. M. Chow,

- and J. M. Gambetta, Experimental demonstration of fault-tolerant state preparation with superconducting qubits, Physical Review Letters 119, 180501 (2017).
- [15] R. Harper and S. T. Flammia, Fault-tolerant logical gates in the ibm quantum experience, Physical Review Letters 122, 080504 (2019).
- [16] N. M. Linke, M. Gutierrez, K. A. Landsman, C. Figgatt, S. Debnath, K. R. Brown, and C. Monroe, Fault-tolerant quantum error detection, Science Advances 3, e1701074 (2017).
- [17] S. Krinner, N. Lacroix, A. Remm, A. Di Paolo, E. Genois, C. Leroux, C. Hellings, S. Lazar, F. Swiadek, J. Herrmann, G. J. Norris, C. K. Andersen, M. Müller, A. Blais, C. Eichler, and A. Wallraff, Realizing repeated quantum error correction in a distance-three surface code, Nature 605, 669 (2022).
- [18] Google Quantum AI and collaborators, Suppressing quantum errors by scaling a surface code logical qubit, Nature 614, 676 (2023).
- [19] D. Bluvstein, S. J. Evered, A. A. Geim, S. H. Li, H. Zhou, T. Manovitz, S. Ebadi, M. Cain, M. Kalinowski, D. Hangleiter, J. P. Bonilla Ataides, N. Maskara, I. Cong, X. Gao, P. Sales Rodriguez, T. Karolyshyn, G. Semeghini, M. J. Gullans, M. Greiner, V. Vuletić, and M. D. Lukin, Logical quantum processor based on reconfigurable atom arrays, Nature 626, 58 (2023).
- [20] L. Postler, F. Butt, I. Pogorelov, C. D. Marciniak, S. Heußen, R. Blatt, P. Schindler, M. Rispler, M. Müller, and T. Monz, Demonstration of fault-tolerant steane quantum error correction, PRX Quantum 5, 030326 (2024).
- [21] Google Quantum AI and collaborators, Quantum error correction below the surface code threshold (2024), arXiv:2408.13687 [quant-ph].
- [22] B. W. Reichardt, D. Aasen, R. Chao, A. Chernoguzov, W. van Dam, J. P. Gaebler, D. Gresh, D. Lucchetti, M. Mills, S. A. Moses, B. Neyenhuis, A. Paetznick, A. Paz, P. E. Siegfried, M. P. da Silva, K. M. Svore,

- Z. Wang, and M. Zanner, Demonstration of quantum computation and error correction with a tesseract code (2024), arXiv:2409.04628 [quant-ph].
- [23] Google Quantum AI and collaborators, Scaling and logic in the color code on a superconducting quantum processor (2024), arXiv:2412.14256 [quant-ph].
- [24] Google Quantum AI and collaborators, Demonstrating dynamic surface codes (2024), arXiv:2412.14360 [quantph].
- [25] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, Elucidating reaction mechanisms on quantum computers, Proceedings of the National Academy of Sciences, 201619152 (2017).
- [26] C. Gidney and M. Ekera, How to factor 2048 bit rsa integers in 8 hours using 20 million noisy qubits, Quantum 5, 433 (2021).
- [27] A. M. Dalzell, S. McArdle, M. Berta, P. Bienias, C.-F. Chen, A. Gilyén, C. T. Hann, M. J. Kastoryano, E. T. Khabiboulline, A. Kubica, G. Salton, S. Wang, and F. G. S. L. Brandão, Quantum algorithms: A survey of applications and end-to-end complexities (2023), arXiv:2310.03011 [quant-ph].
- [28] R. Alicki, D. A. Lidar, and P. Zanardi, Internal consistency of fault-tolerant quantum error correction in light of rigorous derivations of the quantum Markovian limit, Phys. Rev. A 73, 052311 (2006).
- [29] J. Preskill, Sufficient condition on noise correlations for scalable quantum computing, Quant. Inf. Comput. 13, 181 (2013).
- [30] B. D. Clader, C. J. Trout, J. P. Barnes, K. Schultz, G. Quiroz, and P. Titum, Impact of correlations and heavy tails on quantum error correction, Physical Review A 103, 052428 (2021).
- [31] A. Sriram, N. O'Dea, Y. Li, T. Rakovszky, and V. Khemani, Non-uniform noise rates and griffiths phases in topological quantum error correction (2024), arXiv:2409.03325 [quant-ph].
- [32] J. F. Kam, S. Gicev, K. Modi, A. Southwell, and M. Usman, Detrimental non-markovian errors for surface code memory (2024), arXiv:2410.23779 [quant-ph].
- [33] E. Knill, R. Laflamme, Concatenated Quantum Codes (1996), quant-ph/9608012.
- [34] P. Aliferis, D. Gottesman, and J. Preskill, Quantum accuracy threshold for concatenated distance-3 codes, Quant. Inf. Comput. 6, 97 (2006).
- [35] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, Topological quantum memory, Journal of Mathematical Physics 43, 4452 (2002).
- [36] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, Phys. Rev. A 86, 032324 (2012).
- [37] H. Bombin and M. A. Martin-Delgado, Topological quantum distillation, Physical Review Letters 97, 180501 (2006).
- [38] N. P. Breuckmann and J. N. Eberhardt, Quantum lowdensity parity-check codes, PRX Quantum 2, 040101 (2021).
- [39] L. Viola and S. Lloyd, Dynamical suppression of decoherence in two-state quantum systems, Phys. Rev. A 58, 2733 (1998).
- [40] L. Viola, E. Knill, and S. Lloyd, Dynamical decoupling of open quantum systems, Phys. Rev. Lett. 82, 2417 (1999).
- [41] L.-M. Duan and G.-C. Guo, Suppressing environmental

- noise in quantum computation through pulse control, Physics Letters A **261**, 139 (1999).
- [42] D. Vitali and P. Tombesi, Using parity kicks for decoherence control, Physical Review A 59, 4178 (1999).
- [43] V. Tripathi, N. Goss, A. Vezvaee, L. B. Nguyen, I. Siddiqi, and D. A. Lidar, Qudit dynamical decoupling on a superconducting quantum processor, Phys. Rev. Lett. 134, 050601 (2025).
- [44] B. Pokharel, N. Anand, B. Fortman, and D. A. Lidar, Demonstration of fidelity improvement using dynamical decoupling with superconducting qubits, Phys. Rev. Lett. 121, 220502 (2018).
- [45] A. M. Souza, Process tomography of robust dynamical decoupling with superconducting qubits, Quantum Information Processing 20, 10.1007/s11128-021-03176-z (2021).
- [46] N. Ezzell, B. Pokharel, L. Tewala, G. Quiroz, and D. A. Lidar, Dynamical decoupling for superconducting qubits: A performance survey, Phys. Rev. Appl. 20, 064027 (2023).
- [47] C. Tong, H. Zhang, and B. Pokharel, Empirical learning of dynamical decoupling on quantum processors (2024), arXiv:2403.02294 [quant-ph].
- [48] A. Seif, H. Liao, V. Tripathi, K. Krsulich, M. Malekakhlagh, M. Amico, P. Jurcevic, and A. Javadi-Abhari, Suppressing correlated noise in quantum computers via context-aware compiling, in 2024 ACM/IEEE 51st Annual International Symposium on Computer Architecture (ISCA) (IEEE, 2024) pp. 310–324.
- [49] A. Rahman, D. J. Egger, and C. Arenz, Learning how to dynamically decouple (2024), arXiv:2405.08689 [quantph].
- [50] F. Arute, K. Arva, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. S. L. Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi, B. Foxen, A. Fowler, C. Gidney, M. Giustina, R. Graff, K. Guerin, S. Habegger, M. P. Harrigan, M. J. Hartmann, A. Ho, M. Hoffmann, T. Huang, T. S. Humble, S. V. Isakov, E. Jeffrey, Z. Jiang, D. Kafri, K. Kechedzhi, J. Kelly, P. V. Klimov, S. Knysh, A. Korotkov, F. Kostritsa, D. Landhuis, M. Lindmark, E. Lucero, D. Lyakh, S. Mandrà, J. R. Mc-Clean, M. McEwen, A. Megrant, X. Mi, K. Michielsen, M. Mohseni, J. Mutus, O. Naaman, M. Neeley, C. Neill, M. Y. Niu, E. Ostby, A. Petukhov, J. C. Platt, C. Quintana, E. G. Rieffel, P. Roushan, N. C. Rubin, D. Sank, K. J. Satzinger, V. Smelyanskiy, K. J. Sung, M. D. Trevithick, A. Vainsencher, B. Villalonga, T. White, Z. J. Yao, P. Yeh, A. Zalcman, H. Neven, and J. M. Martinis, Quantum supremacy using a programmable superconducting processor, Nature 574, 505 (2019).
- [51] P. Jurcevic, A. Javadi-Abhari, L. S. Bishop, I. Lauer, D. F. Bogorin, M. Brink, L. Capelluto, O. Günlük, T. Itoko, N. Kanazawa, A. Kandala, G. A. Keefe, K. Krsulich, W. Landers, E. P. Lewandowski, D. T. McClure, G. Nannicini, A. Narasgond, H. M. Nayfeh, E. Pritchett, M. B. Rothwell, S. Srinivasan, N. Sundaresan, C. Wang, K. X. Wei, C. J. Wood, J.-B. Yau, E. J. Zhang, O. E. Dial, J. M. Chow, and J. M. Gambetta, Demonstration of quantum volume 64 on a superconducting quantum computing system, Quantum Sci. Technol. 6, 025020 (2021).

- [52] E. Bäumer, V. Tripathi, D. S. Wang, P. Rall, E. H. Chen, S. Majumder, A. Seif, and Z. K. Minev, Efficient long-range entanglement using dynamic circuits, PRX Quantum 5, 030339 (2024).
- [53] B. Pokharel and D. A. Lidar, Demonstration of algorithmic quantum speedup, Phys. Rev. Lett. 130, 210602 (2023).
- [54] B. Pokharel and D. Lidar, Better-than-classical grover search via quantum error detection and suppression, npj Quantum Information 10, 10.1038/s41534-023-00794-6 (2024).
- [55] P. Singkanipa, V. Kasatkin, Z. Zhou, G. Quiroz, and D. A. Lidar, Demonstration of algorithmic quantum speedup for an abelian hidden subgroup problem (2024), arXiv:2401.07934 [quant-ph].
- [56] E. Bäumer, V. Tripathi, A. Seif, D. Lidar, and D. S. Wang, Quantum fourier transform using dynamic circuits (2024), arXiv:2403.09514 [quant-ph].
- [57] H. K. Ng, D. A. Lidar, and J. Preskill, Combining dynamical decoupling with fault-tolerant quantum computation, Phys. Rev. A 84, 012305 (2011).
- [58] G. A. Paz-Silva and D. A. Lidar, Optimally combining dynamical decoupling and quantum error correction, Scientific Reports 3, 10.1038/srep01530 (2013).
- [59] H. Goto, Y. Ho, and T. Kanao, Measurement-free fault-tolerant logical-zero-state encoding of the distance-three nine-qubit surface code in a one-dimensional qubit array, Phys. Rev. Res. 5, 043137 (2023).
- [60] J.-X. Han, J. Zhang, G.-M. Xue, H. Yu, and G. Long, Protecting logical qubits with dynamical decoupling (2024), arXiv:2402.05604 [quant-ph].
- [61] G. Quiroz and D. A. Lidar, Optimized dynamical decoupling via genetic algorithms, Phys. Rev. A 88, 052306 (2013).
- [62] G. T. Genov, D. Schraft, N. V. Vitanov, and T. Half-mann, Arbitrarily accurate pulse sequences for robust dynamical decoupling, Phys. Rev. Lett. 118, 133202 (2017).
- [63] P. Wocjan, Efficient decoupling schemes with bounded controls based on eulerian orthogonal arrays, Physical Review A 73, 062317 (2006).
- [64] V. Tripathi, H. Chen, M. Khezri, K.-W. Yip, E. Levenson-Falk, and D. A. Lidar, Suppression of crosstalk in superconducting qubits using dynamical decoupling, Phys. Rev. Appl. 18, 024068 (2022).
- [65] Z. Zhou, R. Sitler, Y. Oda, K. Schultz, and G. Quiroz, Quantum crosstalk robust quantum control, Phys. Rev. Lett. 131, 210802 (2023).
- [66] B. Evert, Z. G. Izquierdo, J. Sud, H.-Y. Hu, S. Grabbe, E. G. Rieffel, M. J. Reagor, and Z. Wang, Syncopated dynamical decoupling for suppressing crosstalk in quantum circuits (2024), arXiv:2403.07836 [quant-ph].
- [67] A. F. Brown and D. A. Lidar, Efficient chromaticnumber-based multi-qubit decoherence and crosstalk suppression (2024), arXiv:2406.13901.
- [68] E. Knill and R. Laflamme, Theory of quantum errorcorrecting codes, Phys. Rev. A 55, 900 (1997).
- [69] J. Ku, X. Xu, M. Brink, D. C. McKay, J. B. Hertzberg, M. H. Ansari, and B. L. T. Plourde, Suppression of unwanted ZZ interactions in a hybrid two-qubit system, Phys. Rev. Lett. 125, 200504 (2020).
- [70] K. X. Wei, E. Magesan, I. Lauer, S. Srinivasan, D. F. Bogorin, S. Carnevale, G. A. Keefe, Y. Kim, D. Klaus, W. Landers, N. Sundaresan, C. Wang, E. J. Zhang,

- M. Steffen, O. E. Dial, D. C. McKay, and A. Kandala, Hamiltonian engineering with multicolor drives for fast entangling gates and quantum crosstalk cancellation, Physical Review Letters 129, 060501 (2022).
- [71] P. Zanardi, Symmetrizing evolutions, Physics Letters A 258, 77 (1999).
- [72] P. Zanardi, Stabilizing quantum information, Physical Review A 63, 012301 (2000).
- [73] A. A. Maudsley, Modified carr-purcell-meiboom-gill sequence for nmr fourier imaging applications, Journal of Magnetic Resonance (1969) 69, 488 (1986).
- [74] K. Khodjasteh and D. A. Lidar, Fault-tolerant quantum dynamical decoupling, Physical Review Letters 95, 180501 (2005).
- [75] G. S. Uhrig, Keeping a quantum bit alive by optimized π-pulse sequences, Phys. Rev. Lett. 98, 100504 (2007).
- [76] J. R. West, B. H. Fong, and D. A. Lidar, Near-optimal dynamical decoupling of a qubit, Phys. Rev. Lett. 104, 130501 (2010).
- [77] Z.-Y. Wang and R.-B. Liu, Protection of quantum systems by nested dynamical decoupling, Phys. Rev. A 83, 022306 (2011).
- [78] Y. Xia, G. S. Uhrig, and D. A. Lidar, Rigorous performance bounds for quadratic and nested dynamical decoupling, Phys. Rev. A 84, 062332 (2011).
- [79] J. E. Gough and H. I. Nurdin, Can quantum markov evolutions ever be dynamically decoupled?, in 2017 IEEE 56th Annual Conference on Decision and Control (CDC) (2017) pp. 6155–6160.
- [80] K. Szczygielski and R. Alicki, Markovian theory of dynamical decoupling by periodic control, Physical Review A 92, 022349 (2015).
- [81] C. Addis, F. Ciccarello, M. Cascio, G. M. Palma, and S. Maniscalco, Dynamical decoupling efficiency versus quantum non-markovianity, New Journal of Physics 17, 123004 (2015).
- [82] C. Arenz, D. Burgarth, P. Facchi, and R. Hillier, Dynamical decoupling of unbounded hamiltonians, *Journal of Mathematical Physics*, Journal of Mathematical Physics 59, 032203 (2018).
- [83] E. Mozgunov and D. Lidar, Completely positive master equation for arbitrary driving and small level spacing, Quantum 4, 227 (2020).
- [84] K. Khodjasteh and D. A. Lidar, Quantum computing in the presence of spontaneous emission by a combined dynamical decoupling and quantum-error-correction strategy, Physical Review A 68, 022322 (2003), erratum: ibid, Phys. Rev. A 72, 029905 (2005).
- [85] P. Aliferis, F. Brito, D. P. DiVincenzo, J. Preskill, M. Steffen, and B. M. Terhal, Fault-tolerant computing with biased-noise superconducting qubits: a case study, New Journal of Physics 11, 013061 (2009).
- [86] J. P. Bonilla Ataides, D. K. Tuckett, S. D. Bartlett, S. T. Flammia, and B. J. Brown, The xzzx surface code, Nature Communications 12, 2172 (2021).
- [87] L. Viola, E. Knill, and S. Lloyd, Dynamical generation of noiseless quantum subsystems, Phys. Rev. Lett. 85, 3520 (2000).
- [88] M. S. Byrd and D. A. Lidar, Comprehensive encoding and decoupling solution to problems of decoherence and design in solid-state quantum computing, Phys. Rev. Lett. 89, 047901 (2002).
- [89] D. A. Lidar, Towards fault tolerant adiabatic quantum computation, Phys. Rev. Lett. **100**, 160506 (2008).

- [90] G. Quiroz, B. Pokharel, J. Boen, L. Tewala, V. Tripathi, D. Williams, L.-A. Wu, P. Titum, K. Schultz, and D. Lidar, Dynamically generated decoherence-free subspaces and subsystems on superconducting qubits, Reports on Progress in Physics 87, 097601 (2024).
- [91] D. Gottesman, Stabilizer Codes and Quantum Error Correction, Ph.D. thesis, California Institute of Technology, Pasadena, CA (1997), quant-ph/9705052.
- [92] R. Cleve and D. Gottesman, Efficient computations of encodings for quantum error correction, Physical Review A 56, 76 (1997).
- [93] O. Higgott, M. Wilson, J. Hefford, J. Dborin, F. Hanif, S. Burton, and D. E. Browne, Optimal local unitary encoding circuits for the surface code, Quantum 5, 517 (2021).
- [94] D. C. McKay, C. J. Wood, S. Sheldon, J. M. Chow, and J. M. Gambetta, Efficient z gates for quantum computing, Physical Review A 96, 022330 (2017).
- [95] A. Vezvaee, V. Tripathi, D. Kowsari, E. Levenson-Falk, and D. A. Lidar, Virtual z gates and symmetric gate compilation (2024), arXiv:2407.14782 [quant-ph].
- [96] IBM Quantum.
- [97] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, Charge-insensitive qubit design derived from the Cooper pair box, Physical Re-

- view A 76, 042319 (2007).
- [98] A. Blais, A. L. Grimsmo, S. M. Girvin, and A. Wallraff, Circuit quantum electrodynamics, Rev. Mod. Phys. 93, 025005 (2021).
- [99] V. Tripathi, D. Kowsari, K. Saurav, H. Zhang, E. M.Levenson-Falk, and D. A. Lidar, Benchmarking quantum gates and circuits (2024), arXiv:2407.09942 [quant-ph].
- [100] J. Zhang, Z.-Y. Chen, Y.-J. Wang, B.-H. Lu, H.-F. Zhang, J.-N. Li, P. Duan, Y.-C. Wu, and G.-P. Guo, Demonstrating a universal logical gate set in error-detecting surface codes on a superconducting quantum processor (2024), arXiv:2405.09035 [quant-ph].
- [101] B. Hetényi and J. R. Wootton, Creating entangled logical qubits in the heavy-hex lattice with topological codes, PRX Quantum 5, 040334 (2024).
- [102] L. Vaidman, L. Goldenberg, and S. Wiesner, Error prevention scheme with four particles, Physical Review A 54, R1745 (1996).
- [103] D. F. V. James, P. G. Kwiat, W. J. Munro, and A. G. White, Measurement of qubits, Physical Review A 64, 052312 (2001).
- [104] J. Stehlik, D. M. Zajac, D. L. Underwood, T. Phung, J. Blair, S. Carnevale, D. Klaus, G. A. Keefe, A. Carniol, M. Kumph, M. Steffen, and O. E. Dial, Tunable coupling architecture for fixed-frequency transmon superconducting qubits, Phys. Rev. Lett. 127, 080505 (2021).