Lecture 1

My main goal for the next two lectures is to convince you that it should be possible to build a quantum computer in the real world, or at least in some reasonable theoretical model of it. Why might it not be possible? For example, quantum mechanics could turn out to be false. However, the main problem I am going to try to address is noise. A single failed gate in a large circuit could potentially corrupt the entire output. In a very large circuit, the probability of some failure in it approaches one. Therefore perhaps it just isn't possible to factor any number larger than 15. I'm going to argue that this isn't what happens.

Why is noise a particular problem for quantum circuits? Well, what do we know about qubits?

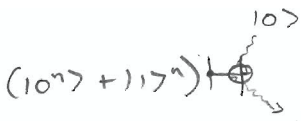
Qubits: tiny

They're really small, like single electrons or photons. They're fragile, easy to lose track of, hard to manipulate... But this isn't a good argument. The reason qubits are small is because of noise — larger systems decohere faster — so this is circular. More fundamentally, qubits are continuous.

continuous $\alpha|0\rangle + \beta|1\rangle$

At the very least, then, there will always be precision errors. Even more fundamentally, qubits are entangled

entangled $\frac{1}{\sqrt{2}}(|0^n\rangle + |1^n\rangle)$



$\frac{1}{2}|0^n\rangle \quad \frac{1}{2}|1^n\rangle$

For example, Schrödinger's cat state is an equal superposition between being dead and alive. If you look at it, it will collapse to being either dead or alive, but in fact it is even worse than that. Take a single photon, up here, have it interact with just one of the qubits, say the last one with a CNOT, before bouncing away. Then the state becomes $|0^{n+1}\rangle + |1^{n+1}\rangle$ and after tracing out the photon it collapses. Entanglement is very fragile! We'd like to build up highly entangled states in our quantum computer, but we don't want a single stray photon to be able to crash the computer.

I'm going to show how we can protect our quantum computer against these kinds of events.

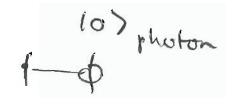
My main goal is to convince you that building a working quantum computer is in principle possible. Why might it not be? Well, noise.

Qubits are these little tiny

- small (fragile) e^-
- continuous - gates will always be slightly off
- entangled

$$\frac{1}{\sqrt{2}}\left(|000\cdots0\rangle + |111\cdots1\rangle\right)$$

dead cat   live cat



$\frac{1}{2}: 0^n$ $\frac{1}{2}\ 1^n$ mixture

qu vs. classical

disadvantages: ~~no cloning~~

have to protect phase, not just bit flips (~~even continuous errors~~)

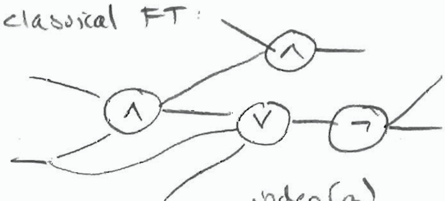
quantum codes are more complicated, harder to encode

advantages:

classical FT $\Rightarrow$ can assume classical computer to help us out

qu teleportation

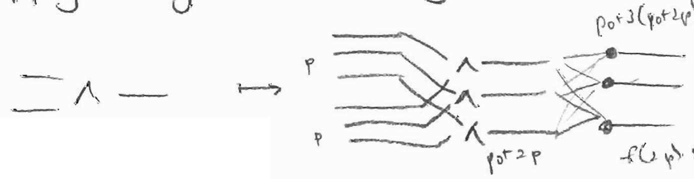
classical FT:

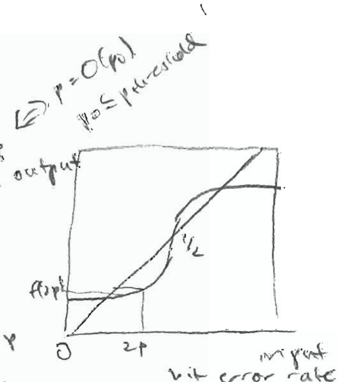


encode $0 \mapsto 0^{n \approx \log N}$

$1 \mapsto 1^n$

apply each gate transversally





physical systems have additional locality constraint, which we'll ignore

Von Neumann, Gács

* Ike's NIST talk
detected err threshold 1/2
my Banff talk!
using large random codes
+
Rains bound

Noise

Markovian noise

$V^\dagger V = I_S$

$V|\psi\rangle = U(|\psi\rangle_S |0\rangle_E)$

"Markovian" b/c env. refreshed after each event, no memory

non-Markovian 

need to track combined state ρ_{SE}

$$\mathcal{E}(\rho) = tr_E(V\rho V^\dagger) = \sum_k {}_E\langle k|(V\rho V^\dagger)|k\rangle_E$$

$$= \sum_k E_k \rho E_k^\dagger \quad \text{"operator sum repn"}$$

$$Tr\,\rho = Tr\,\mathcal{E}(\rho) = \sum_k Tr\, E_k^\dagger E_k \rho \quad \Rightarrow \sum_k E_k^\dagger E_k = I$$

$$= Tr\left(\sum_k E_k^\dagger E_k\right)\rho$$

(plug in $\rho = |i\rangle\langle j|$, $tr\,\rho = \delta_{ij} = \langle j|\sum_k E_k^\dagger E_k|i\rangle$)

... Pauli's ...

$$V = \sum_k E_k \otimes |k\rangle_E \qquad E_k = \sum_i c_k^i \sigma_i$$

$$= \sum_i \sigma_i \otimes \underbrace{\sum_k c_k^i |k\rangle_E}_{|\sigma^i\rangle_E}$$

* $|\psi\rangle \xrightarrow[\text{noise}]{\text{non-Markov}} \sum_i \sigma_i|\psi\rangle \otimes |\sigma^i\rangle_E$

classically, suff. to protect against X noise

quantumly, " " X, Y, Z

examples:

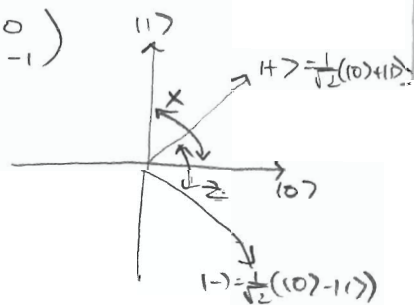
Noise Markovian noise (S) —isometry→ (S)(B) —tre→ (S)(S)

non Markov. noise (S)(E) → (S)(E), E ∞ dim, etc.

Paulis

$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ $Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$ $Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

τ_0 σ_1 τ_x σ_2 τ_y σ_3 τ_z



- multiplication table $\sigma_j \sigma_k = i \varepsilon_{jkl} \sigma_l$

$XY = iZ = -YX$

$YZ = iX = -ZY$

$ZX = iY = -XZ$

- Paulis form a basis over $\mathbb{C}$ for all 2×2 matrices

over $\mathbb{R}$ for Hermitian ″

→ "operator sum"

$$\mathcal{E}(\rho) = \mathrm{tr}_E\left[U(\rho \otimes \rho_E)U^\dagger\right] = \sum_k \underbrace{{}_E\langle k|U|0\rangle_E}_{E_k}\, \rho\, \underbrace{{}_E\langle 0|U^\dagger|k\rangle_E}_{E_k^\dagger}$$

$\sum_k E_k^\dagger E_k = I$ trace preserving

(in fact, any completely positive trace-preserving map can be so expanded)

given $\{E_k\}$, $E_k = \sum_i c_{ik}\sigma^i$

may take $U = \sum_k E_k \otimes |k\rangle_E$ an isometry

$$= \sum_i \sigma^i \otimes \underbrace{\sum_k c_{ik}|k\rangle_E}_{|i\rangle_E}$$

(*) $|\psi\rangle \xrightarrow{\text{noise}} \sum_i \sigma^i|\psi\rangle \otimes |i \text{ environment}\rangle$

classically, suffices to protect against X noise — also see [NC, p. 443]

quantumly, ″ ″ ″ ″ X, Y, Z noise

examples

1. depolarizing noise

randomize a qubit w/ prob. p

$E_i \propto \sigma_i$

$${}_E\langle\sigma^j|\sigma^i\rangle_E = \delta_{ij} \cdot \begin{cases} p, & i \neq 0 \\ 1-p, & i = 0 \end{cases}$$

$$\begin{pmatrix} \rho_{00} & \rho_{01} \\ \rho_{10} & \rho_{11} \end{pmatrix} \mapsto \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

2. amplitude damping

$(\alpha|0\rangle + \beta|1\rangle) \otimes |0\rangle_E$

$\alpha|0\rangle|0\rangle_E$

$+ \beta\sqrt{1-\gamma}\,|1\rangle|0\rangle_E$

$+ \beta\sqrt{\gamma}\,|0\rangle|1\rangle_E$

$E_0 = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-p} \end{pmatrix}$ $E_1 = \begin{pmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{pmatrix}$

$= 1 - \frac{1-z}{2}(1-\sqrt{1-\gamma})$ $= \frac{\sqrt{\gamma}}{2}(X+iY)$

3. phase damping

= measurement by environment

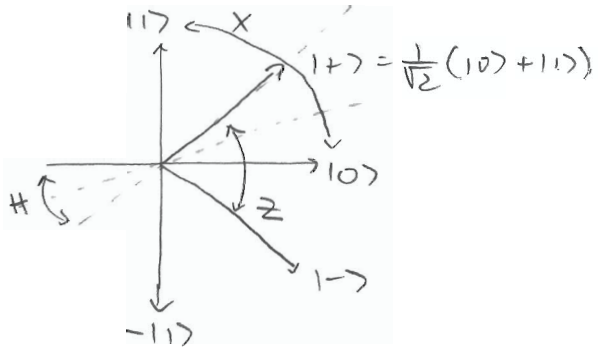
$U = |00\rangle_E\langle 0|0| + \sqrt{1-\gamma}\,|1\rangle|0\rangle_E\langle 1|1| + \sqrt{\gamma}\,|1\rangle|1\rangle_E\langle 1|1|$

$E_0 = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{pmatrix}$ $E_1 = \begin{pmatrix} 0 & 0 \\ 0 & \sqrt{\gamma} \end{pmatrix}$

$\frac{1+\sqrt{1-\gamma}}{2} I + \frac{1-\sqrt{1-\gamma}}{2} Z$ ″ $\frac{\sqrt{\gamma}}{2} I - \frac{\sqrt{\gamma}}{2} Z$

(1-qubit) Paulis

	AKA.
$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	σ_0, σ_I
$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	σ_1, σ_X
$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	σ_2, σ_Y
$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	σ_3, σ_Z



- multiplication

$\sigma^2 = I \qquad XY = iZ = -YX$
$YZ = iX = -ZY$
$ZX = iY = -XZ$

- basis/$\mathbb{C}$ for all $2^n \times 2^n$ matrices

: <noise>

Now I'd like to talk about qo stabilizer codes and the stabilizer formalism. This is all about multi-qubit Pauli operators.

n-qubit Paulis:

- $X \otimes Z \otimes Y \otimes I \otimes I \sim XZY\,II$

These are just tensor products of single-qubit Paulis. It is fairly standard, though, to drop the tensor signs when writing them down.

First of all, as in the 1-qubit case,

- -- form a basis over $\mathbb{C}$ for all $2^n \times 2^n$ matrices (4^n)

(As a sanity check, there are 4^n n-qubit Paulis.) Also, although we won't need it today,

- -- " " $\mathbb{R}$ for Hermitian " "

for stab states → must coordinates are 0 (is not a group)

I'm going to state a sequence of facts now, leaving their proofs as exercises. By the end of today, I hope the proofs will be very easy.

- Ex: $P^2 = I$, half evs +1, half −1

$$\text{Dim Range } \tfrac{1}{2}(1+P) = \text{Dim Range } \tfrac{1}{2}(1-P) = \frac{2^n}{2} = 2^{n-1}$$

Every Pauli squares to the identity. This implies that its eigenvalues are all either +1 or −1. In fact the division is exactly half/half. In other words, $\frac{1}{2}(1+P)$ is the projector onto the +1 eigenspace of P, and the dimension of its range is exactly that of the projector onto the −1 eigenspace, $\frac{1}{2}(1-P)$: $1^n/2$.

- Ex: P, Q either commute or anticommute

$$[P,Q] = PQ - QP = 0 \qquad \{P,Q\} = PQ + QP = 0$$

For any P, Q, either the commutator or the "anticommutator" is exactly zero. In the 1-qubit case you can see that every P commutes with the identity and with itself, but otherwise they anticommute.

Whether P and Q commute or anticommute depends on whether the number of positions where they differ nontrivially is even or odd. Letting

$$\chi(P=\bigotimes_i P_i,\ Q=\bigotimes_i Q_i) = \#\{i : P_i = Q_i;\ P_i, Q_i \neq 1\}$$

$$PQ - (-1)^{\chi} QP = 0$$

For example

$$P = X\ Y\ Z\ X\ Y\ 1\ Y\ Z$$
$$Q = Z\ Y\ X\ 1\ Z\ X\ X\ Y$$

these anticommute, because 5 is odd.

- Ex: $[P,Q]=0$

$$\begin{matrix} +1 & +1 \\ +1 & -1 \\ -1 & +1 \\ -1 & -1 \end{matrix} \quad \leftarrow 2^{n-2}$$

Next exercise: if P and Q are different, nontrivial, and commute, then they can be simultaneously diagonalized. That is, you can break the Hilbert space into the direct sum of four spaces. All vectors for example in the second space are $+1$ eigenvectors of P, and -1 eigenvectors of Q.

$$\left(\dim \text{Range } \tfrac{I}{4}(I \pm P)(I \pm Q) = 2^{n-2}\right)$$

is another way of putting it.

More generally,

- Ex: $P_1, \ldots, P_k$ indep. Paulis commuting

$$\begin{matrix} +1 & +1 & \cdots & +1 \\ +1 & +1 & \cdots & -1 \\ & & \vdots & \\ -1 & -1 & \cdots & -1 \end{matrix} \quad \dim = 2^{n-k}$$

If you have k independent Pauli operators all commuting (pairwise), then they can all be simultaneously diagonalized. Break the Hilbert space up as the direct sum of 2^k subspaces. This second one for example is the intersection of the $+1$ eigenspaces of $P_1, \ldots P_{k-1}$ with the -1 eigenspace of P_k. Then one can show that each of these spaces has equal dimension, 2^{n-k}.

Therefore in a sense these are all equivalent subspaces. Nonetheless it is nice to single one out. The simultaneous $+1$ eigenspace of all the P_i is known as the

"stabilizer subspace" or "code space"

$|\psi\rangle \in$ codespace $\quad P_i|\psi\rangle = |\psi\rangle$ "P_i stabilizes $|\psi\rangle$"

The other spaces, they're all labelled by a string of ± 1's, known as the "syndrome."

I guess this is somewhat abstract, especially as I haven't proved anything. They are all straightforward exercises, though. For example, the first one:

$\exists Q : \{P, Q\} = 0$

$P|\psi\rangle = |\psi\rangle$

$P(Q|\psi\rangle) = -QP|\psi\rangle$

$= -(Q|\psi\rangle)$

I claim take $|\psi\rangle$ stabilized by P. Look at $Q|\psi\rangle$; it's a -1 ev. Q flips the $+1$ and -1 spaces, so they have same dim.

An easier proof might be to argue that w.l.o.g. we can take P to be

$$P = Z \otimes I^{\otimes n-1} = \begin{pmatrix} 1 & & & \\ & \ddots & & \\ & & -1 & \\ & & & \ddots \\ & & & & -1 \end{pmatrix}$$

in which case the statement is obvious.

To make things more concrete, let me give some examples. ← stab. states = codes

The last exercise implies that $k \leq n$. In the case $k=1$, the stabilizer space has dimension 1; it is therefore just a state, known as a stabilizer state

Stabilizer states: $(k=n)$

$n=1$: Z $|0\rangle$ X $|+\rangle$

$-Z$ $|1\rangle$ $-X$ $|-\rangle$

For $n=1$, Z stabilizes $|0\rangle$, $-Z$ stabilizes $|1\rangle$ and so on.

$n=2$: $\{ZI, IX\}$ $|0\rangle \otimes |+\rangle$

$\{ZZ, XX\}$ $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

general n : $\{X^{\otimes n}, Z_i Z_j\}$ $\frac{1}{\sqrt{2}}(|0^n\rangle + |1^n\rangle)$

For $k<n$, the stabilizer subspace has dimension 2^{n-k}, so we can encode, or embed, $n-k$ qubits into it. Some examples.

- $n=1$, $k=0$: trivial code
- $n=2$, $k=1$: $P = ZZ$ span$(|00\rangle, |11\rangle)$ $+1$ syndrome; span$(|01\rangle, |10\rangle)$ -1 "

We can here encode one qubit into the codespace. Any one-bit X error anticommutes with P, so takes you into the -1 syndrome space. This can be detected. On the other hand, single-qubit Z operators commute with P, so act inside the codespace. This code offers no protection against Z errors.

- $n=2$, $k=1$ $P = XX$ span$(|++\rangle, |--\rangle)$

Just the dual..

- n, $k=n-1$ $\{Z_i Z_j\}$ span$(|0^n\rangle, |1^n\rangle)$ ← repetition code

Any classical linear code can be written in stabilizer notation by using Pauli Zs to indicate the parity checks.

The smallest stabilizer code to protect against both bit flips and phase flips, both Xs and Zs, has

- $n=4$, $k=2$

$$\begin{matrix} Z & Z & Z & Z \\ X & X & X & X \end{matrix}$$

EX: eg. ZZZ / XXX anticommute

These operators commute, so they define a 4-dim. codespace holding two qubits. Any Pauli on a single qubit anticommutes with one of the stabilizers, so takes you to an orthogonal syndrome space. We write

$$[\![4, 2, 2]\!]$$

(4: # qubits; 2: # encoded qubits; 2: distance)

The distance is the lowest weight of a nontrivial Pauli which preserves the codespace, i.e., which commutes with the stabilizers. Here, eg.

$$XX11$$

commutes with both stabilizers. The double-bracket notation indicates that it is a qu. code, protecting against both bit & phase flip errs.

The smallest code which can correct an error, not merely detect it, has $n=5$. Its stabilizers are

$$\begin{matrix} X & Z & Z & X & 1 \\ 1 & X & Z & Z & X \\ X & 1 & X & Z & Z \\ Z & X & 1 & X & Z \end{matrix}$$

These are easy to remember because they are all just cyclic permutations of the first one. The last permuted stabilizer,

$$ZZXIX$$

is not independent of the others. To see that this defines a code, one can check that these Paulis all commute with each other.

$$[\![5, 1, 3]\!]$$

So there is one encoded qubit. I'll leave as an exercise to show that the distance is 3 (the lowest weight of any nontrivial operator commuting with all stabilizers is 3). What this means is that any one-qubit error is correctable. As a sanity check, there are

$$5 \times 3 = 15 \text{ different 1-qubit Pauli errors}$$
$$16 \text{ syndromes}$$

"perfect"

It turns out then that every 1-qubit Pauli moves you from the code space to a different, orthogonal syndrome space. Measuring the syndrome therefore tells you what the error was, so you can undo it to move back.

Recall that if you have not a discrete error, but some linear combination of correctable Pauli errors, the state becomes a superposition spread out over the syndrome spaces. Measuring the syndrome projects down to a discrete error which can be corrected.

Finally, the

Steane $[\![7,1,3]\!]$ code

is based on the classical

$[7,4,3]$ Hamming code

This code has parity checks

$$\begin{matrix} 0\,0\,0\,1\,1\,1\,1 \\ 0\,1\,1\,0\,0\,1\,1 \\ 1\,0\,1\,0\,1\,0\,1 \end{matrix}$$

Translating into stabilizer notation, these become

$$\begin{matrix} I\,I\,I\,Z\,Z\,Z\,Z \\ I\,Z\,Z\,I\,I\,Z\,Z \\ Z\,I\,Z\,I\,Z\,I\,Z \end{matrix}$$

Against X bit-flip errors, the distance is 3. Now add the same stabilizers but in the dual basis:

$$\begin{matrix} I\,I\,I\,X\,X\,X\,X \\ I\,X\,X\,I\,I\,X\,X \\ X\,I\,X\,I\,X\,I\,X \end{matrix}$$

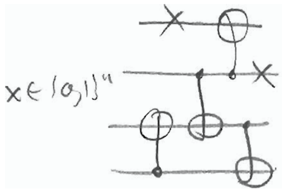
All the Z stabilizers commute with themselves, and the X stabilizers with themselves; to see that this defines a code, one has to check that the X stabilizers commute with the Z ones. This code has distance 3 because any error can be split up into an X error and a Z error, and this code separately corrects one X error and one Z error.

Ex: Derive stabilizer representation for Shor's $[\![9,1,3]\!]$ code — the 3-bit repetition code concatenated on its dual. Show that the distance is 3, and write down generators for the set of logical op's (ie; nontrivial ops within the code/stabilizer space).

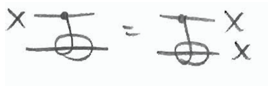
The key to this exercise is to figure out how to concatenate two stabilizer codes. (Either that, or go back a few weeks to when the code was presented.)

Stabilizer algebra "generalization of classical linearity"

$x \in \{0,1\}^n$

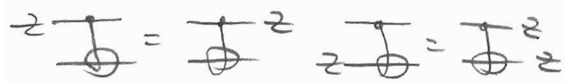


$$y = \begin{pmatrix}1\\1\\0\\0\end{pmatrix} + \begin{pmatrix}1&1&0&0\\0&1&0&0\\0&1&1&1\\0&1&1&0\end{pmatrix} \cdot x$$



$\mathrm{CNOT}(1+a, b) = (1+a, 1+a+b)$
$= \mathrm{CNOT}(a,b) + (1,1)$

$-Z-H- = -H-X-$



<u>Thm</u>: (Gottesman, Knill)

Def: Clifford group $= \{ U : U(\text{Pauli})U^+ \subseteq \text{Pauli} \}$

<u>Ex</u>: $= \langle \mathrm{CNOT},\ H,\ \begin{pmatrix}1&0\\0&i\end{pmatrix} \rangle$

Stabilizer op =
- Clifford unitary
- preparation of $|0\rangle$
- measurement 0/1

Stab. ops are efficiently simulatable.

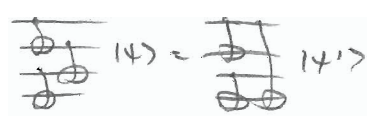
Proof: Claim: States preparable using stabilizer ops are exactly stab states!

1. $Z|0\rangle = |0\rangle$
2. Clifford
3. measurement (just say rule?)

$\langle\psi|0\rangle\langle 0|\psi\rangle = \langle\psi|\frac{1}{2}(1+Q)|\psi\rangle$
$= \langle\psi|\frac{1}{2}(1-Q)|\psi\rangle$

represent state by operators stabilizing it

Examples. 1. 

2. 

2a. cluster states
$|+\rangle\ |0\rangle\ \cdots$
$|0\rangle\ |+\rangle\ \cdots$

2b. how to encode arbitrary stab state?

```
X Z Z X 1
1 X Z Z X      H2        X X X X 1
X 1 X Z Z      H3        [ stuff ]
Z X 1 X Z
Z Z Z Z Z
```

$\xrightarrow{SX_{L34}}$ X 1 1 1 1 [stuff...] $O(n^2)$ gates

$\frac{n^2}{\log n}$ tight

3. teleportation

4. 4-qubit FT syndrome extraction

```
X X X X
Z Z Z Z
X X . .
. Z . Z
X . X .
. . Z Z
```

$$\rho(\{c_\sigma\}) = \frac{1}{2^n}\sum_\sigma c_\sigma \sigma$$

$\downarrow$

$$\rho_L = \frac{1}{2^n}\Big(\sum_{r\in S}\sigma\Big)\Big(\sum_{\sigma_L} c_\sigma \sigma_L\Big)$$

2c. torus code

```
Z Z X X
X Z X X
X X Z Z
```