

Overview of TQEC code

2025/02/05 – Adrien Suau

The main module: tqec – Files

src/tqec/ => Main code directory

- ├─ circuit
- ├─ _cli
- ├─ compile
- ├─ computation
- ├─ gallery
- ├─ __init__.py => Requirement to make tqec a Python module
- ├─ interop
- ├─ plaqueette
- ├─ py.typed => Flag file to say that tqec is fully typed
- ├─ simulation
- ├─ templates
- ├─ utils
- └─ _version.py => Contains tqec version

The main module: tqec – Auxiliary sub-modules

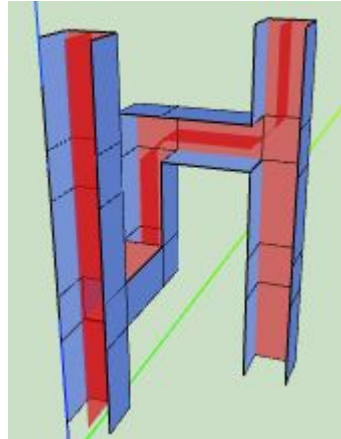
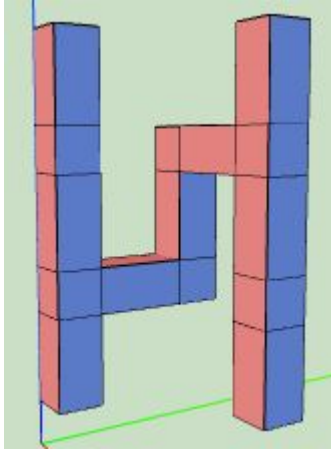
src/tqec/	=> Main code directory
— circuit	
— _cli	=> Initial implementation of tqec CLI
— compile	
— computation	
— gallery	=> Implementation of a few standard computations (memory , CNOT , etc.)
— __init__.py	
— interop	=> Interoperability code (standard colours , import from SketchUp)
— plaqueette	
— py.typed	
— simulation	=> Utilities related to simulations using sinter (starting & plotting)
— templates	
— utils	=> Various utilities (Position2D , LinearFunction , NoiseModel , etc.)
— _version.py	

The main module: tqec – Main sub-modules

src/tqec/	=> Main code directory
— circuit	=> Implementation of ScheduledCircuit
— _cli	
— compile	=> BlockGraph => CompiledGraph => ScheduledCircuit => stim.Circuit
— computation	=> BlockGraph representation of a computation
— gallery	
— __init__.py	
— interop	
— plaquette	=> Representation of a local quantum circuit: Plaquette
— py.typed	
— simulation	
— templates	=> Representation of a 2-dimensional, scalable, arrangement of plaquettes
— utils	
— _version.py	

Let's start somewhere: the computation

How to represent:

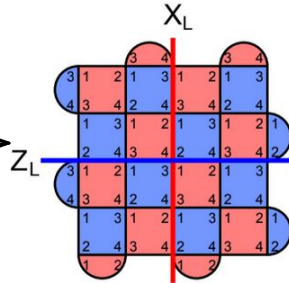
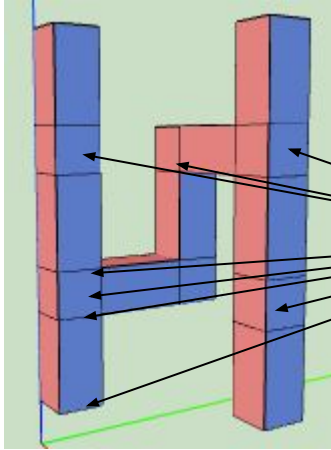


Using [tqec.computation](#):

- [BlockGraph](#)
- [CorrelationSurface](#)

Going into details: layers, templates and plaquettes

Using [tqec.templates](#) and [tqec.plaquette](#):



1	5	6	5	6	2
7	9	10	9	10	11
8	10	9	10	9	12
7	9	10	9	10	11
8	10	9	10	9	12
3	13	14	13	14	4

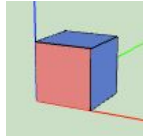
=> [Template](#)

```
{  
    6: Plaquette(...),  
    7: Plaquette(...),  
    9: Plaquette(...),  
    10: Plaquette(...),  
    12: Plaquette(...),  
    13: Plaquette(...),  
    default: Plaquette.empty(),  
}
```

=> [Plaquettes](#)

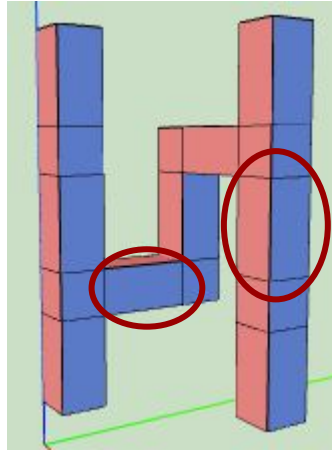
From high-level to low-level

How to translate



to a circuit? $\Rightarrow$ tqec.compile.specs and [BlockBuilder](#).

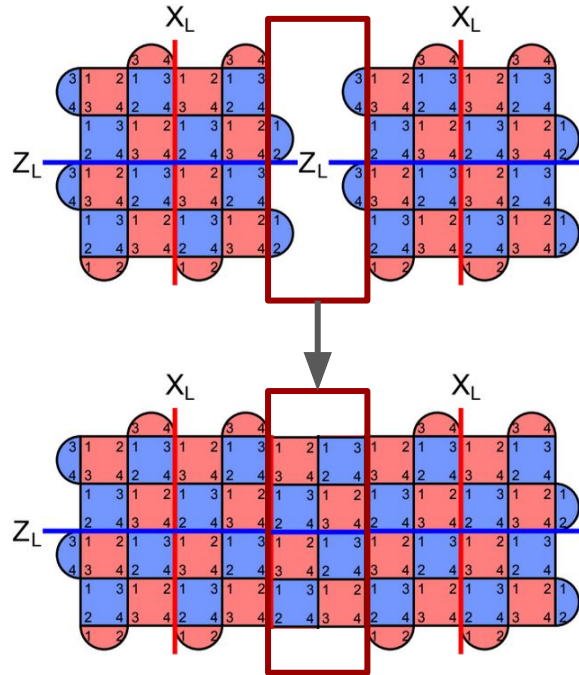
What about junctions?



$\Rightarrow$ [SubstitutionBuilder](#).

What is a Substitution?

How are 2 logical qubits merged?



A Substitution encodes all the information needed to perform that operation.

Implementing spatial junctions

The main PR is [Tentative implementation of spatial junctions #459](#).

Main issues remaining:

1. One test is failing and needs to be properly resolved.
2. Implement the plaquettes for spatial junction (see [#461](#)).
3. Allow alternating plaquettes ([#460](#) should allow that).
4. Allow overlapping of plaquettes in time([#462](#)).

How to contribute:

- Assign yourself to the issue you want to work on.
- Branch from [#459](#) ([feat/spatial_junctions](#)):
`git checkout feat/spatial_junctions`
`git checkout -b [new branch name]`
- Open a PR on [feat/spatial_junctions](#).

