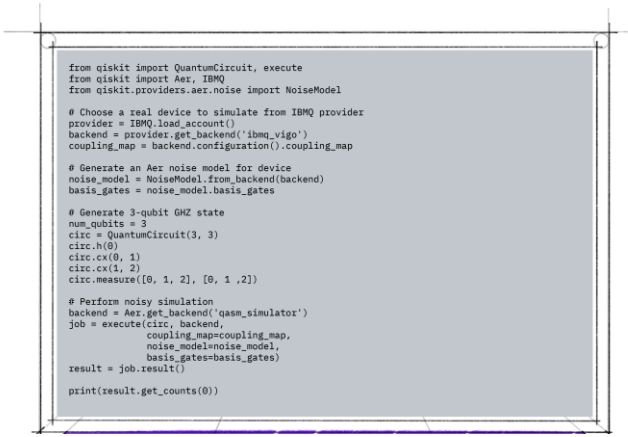
# Good First Issues in Qiskit-Terra

Tharrmashastha SAPV

Mentor: Luciano Bello



```
from qiskit import QuantumCircuit, execute
from qiskit import Aer, IBMQ
from qiskit.providers.aer.noise import NoiseModel

# Choose a real device to simulate from IBMQ provider
provider = IBMQ.load_account()
backend = provider.get_backend('ibmq_vigo')
coupling_map = backend.configuration().coupling_map

# Generate an Aer noise model for device
noise_model = NoiseModel.from_backend(backend)
basis_gates = noise_model.basis_gates

# Generate 3-qubit GHZ state
num_qubits = 3
circ = QuantumCircuit(3, 3)
circ.h(0)
circ.cx(0, 1)
circ.cx(1, 2)
circ.measure([0, 1, 2], [0, 1 ,2])

# Perform noisy simulation
backend = Aer.get_backend('qasm_simulator')
job = execute(circ, backend,
              coupling_map=coupling_map,
              noise_model=noise_model,
              basis_gates=basis_gates)
result = job.result()

print(result.get_counts(0))
```

# Goal of the Project

Fix as many Qiskit Terra issues in Github as possible!!!

# Current Progress

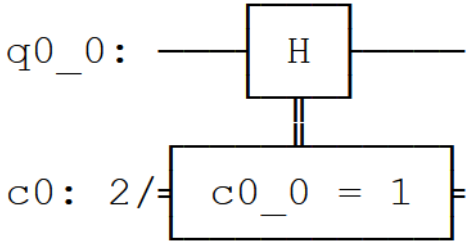
Qiskit

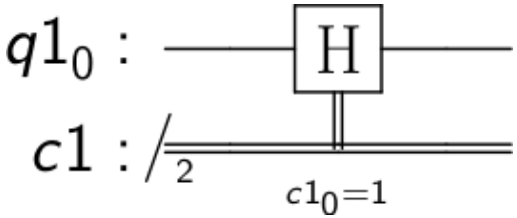
- Issue 1160 – Classical conditioning on single classical bits.
  - Single bit conditioning added to instruction class.
  - Added various tests.
  - Errors caused –
    - circuit_to_dag and composing circuits break.
    - circuit drawers break.
    - circuit_to_qasm breaks.
- Issue 3202 – Latex support for complex custom instructions.
- Issue 2092 – Numbering active wires in a multi-qubit gate in latex.

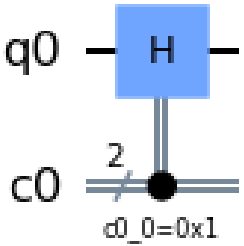
# Bit Conditioning

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-3-76da3468ac34> in <module>
      5 qc = QuantumCircuit(1,2)
      6 qc.h(0).c_if(c[0],1)
----> 7 qc.draw('latex')

~/qiskit_testing/qiskit_terra/test_repo/issue1160/qiskit-terra/qiskit/circuit/quantumcircuit.py in draw(self, output,
scale, filename, style, interactive, plot_barriers, reverse_bits, justify, vertical_compression, idle_wires, with_lay
out, fold, ax, initial_state, cregbundle)
   1484                                 ax=ax,
   1485                                 initial_state=initial_state,
-> 1486                                 cregbundle=cregbundle)
   1487
   1488     def size(self):

~/qiskit_testing/qiskit_terra/test_repo/issue1160/qiskit-terra/qiskit/visualization/circuit_visualization.py in circu
it_drawer(circuit, scale, filename, style, output, interactive, plot_barriers, reverse_bits, justify, vertical_compre
ssion, idle_wires, with_layout, fold, ax, initial_state, cregbundle)
    209                                     with_layout=with_layout,
    210                                     initial_state=initial_state,
--> 211                                     cregbundle=cregbundle)
    212     elif output == 'latex_source':
    213         return _generate_latex_source(circuit,

~/qiskit_testing/qiskit_terra/test_repo/issue1160/qiskit-terra/qiskit/visualization/circuit_visualization.py in _late
x_circuit_drawer(circuit, scale, filename, plot_barriers, reverse_bits, justify, idle_wires, with_layout, initial_sta
te, cregbundle)
```

```
~/qiskit_testing/qiskit_terra/test_repo/issue1160/qiskit-terra/qiskit/converters/circuit_to_dag.py in circuit_to_dag
(circuit)
     60
     61     for instruction, qargs, cargs in circuit.data:
---> 62         dagcircuit.apply_operation_back(instruction.copy(), qargs, cargs)
     63
     64     dagcircuit.duration = circuit.duration

~/qiskit_testing/qiskit_terra/test_repo/issue1160/qiskit-terra/qiskit/dagcircuit/dagcircuit.py in apply_operation_bac
k(self, op, qargs, cargs, condition)
    429         all_cbits = set(all_cbits).union(cargs)
    430
--> 431         self._check_condition(op.name, op.condition)
    432         self._check_bits(qargs, self.output_map)
    433         self._check_bits(all_cbits, self.output_map)

~/qiskit_testing/qiskit_terra/test_repo/issue1160/qiskit-terra/qiskit/dagcircuit/dagcircuit.py in _check_condition(se
lf, name, condition)
    330         """
    331         if (condition is not None and condition[0] not in self.clbits and
--> 332                 condition[0].name not in self.cregs):
    333             raise DAGCircuitError("invalid creg in condition for %s" % name)
    334

AttributeError: 'Clbit' object has no attribute 'name'
```

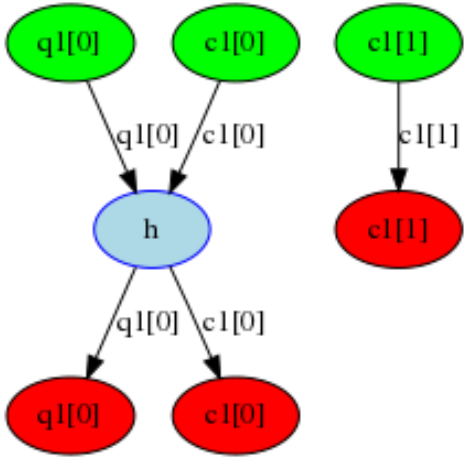


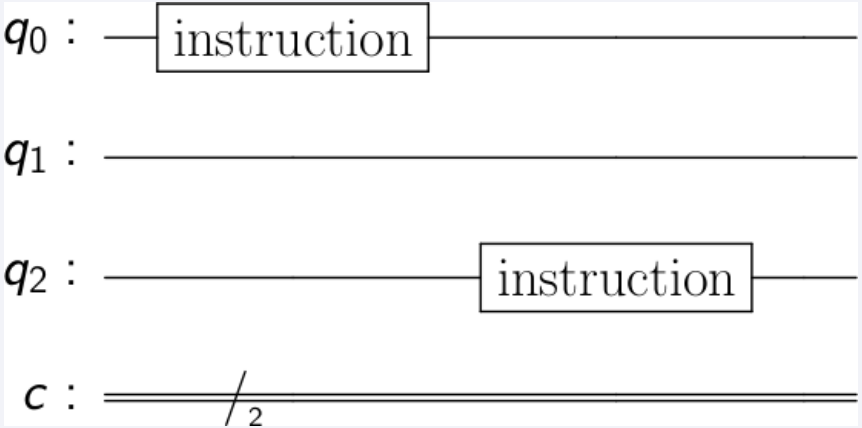



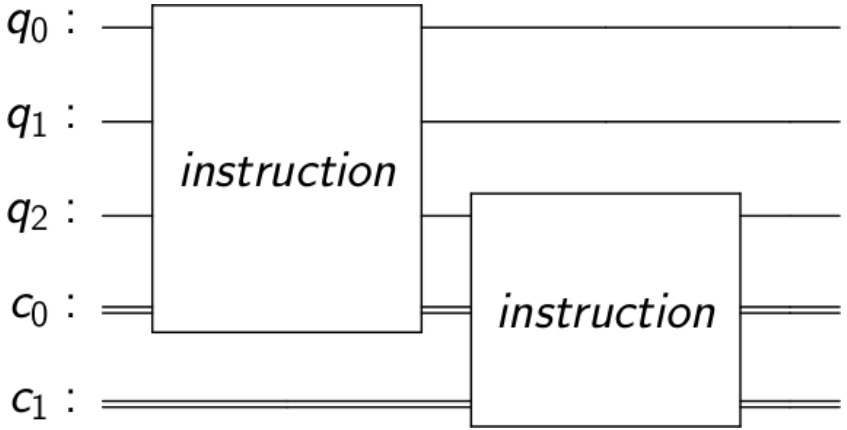
# Bit Conditioning

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-1-dbfab0f40242> in <module>
     10 from qiskit.visualization import dag_drawer
     11
---> 12 dag_drawer(circuit_to_dag(qc))

~/qiskit_testing/qiskit_terra/test_repo/issue3765/qiskit-terra/qiskit/converters/circuit_to_dag.py in circuit_to_dag(circui
t)
     56
     57     for instruction, qargs, cargs in circuit.data:
---> 58         dagcircuit.apply_operation_back(instruction.copy(), qargs, cargs)
     59
     60     dagcircuit.duration = circuit.duration

~/qiskit_testing/qiskit_terra/test_repo/issue3765/qiskit-terra/qiskit/dagcircuit/dagcircuit.py in apply_operation_back(sel
f, op, qargs, cargs, condition)
    365         cargs = cargs or []
    366
--> 367         all_cbits = self._bits_in_condition(op.condition)
    368         all_cbits = set(all_cbits).union(cargs)
    369

~/qiskit_testing/qiskit_terra/test_repo/issue3765/qiskit-terra/qiskit/dagcircuit/dagcircuit.py in _bits_in_condition(self,
cond)
    309             list[Clbit]: list of classical bits
    310         """
--> 311         return [] if cond is None else list(cond[0])
    312
    313     def _add_op_node(self, op, qargs, cargs):

TypeError: 'Clbit' object is not iterable
```

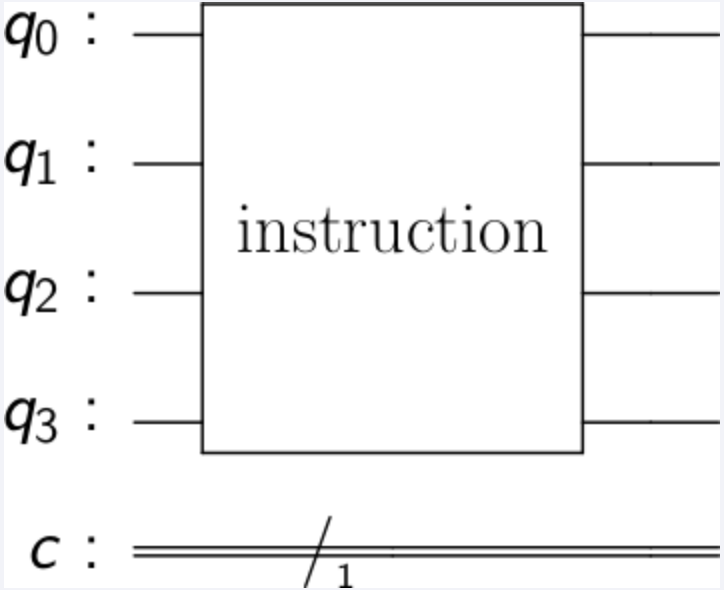


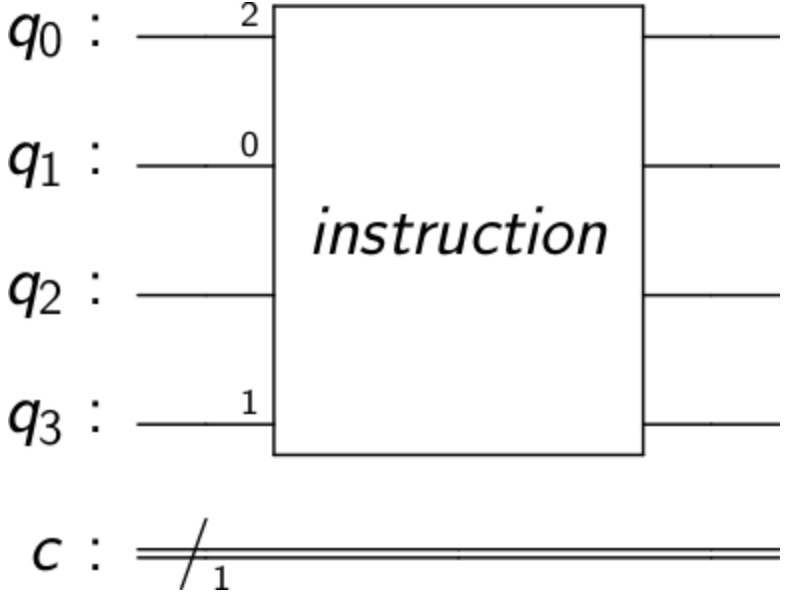
# Issue 3202



# Issue 2092



# Future Works

- Support for single bit conditioning in registerless circuits.
- Work on multiple issues in Visualization and Tools.
  - Adding tests for various visualization modules.
  - Complex custom instruction support in mpl drawer.
- Issues related to transpilers.
- Issues beyond 'Good First Issues'.

Qiskit

# Thank You