quantum_1

March 19, 2020

[2]:
```
import cirq
import numpy as np
from cirq import Circuit
from cirq.devices import GridQubit

length = 3 #number of qbits=9, required=5
qbits = [cirq.GridQubit(x,y) for x in range(length) for y in range(length)]
print(qbits)
circuit = cirq.Circuit()
circuit.append(cirq.H(q) for q in qbits[:5]) #applying hadamard op to all 5␣
 ↪qbits
# print(circuit)
circuit.append(cirq.CNOT(qbits[x],qbits[x+1]) for x in range(4)) #cnot gate on␣
 ↪consecutive qbits
# print(circuit)

circuit.append(cirq.SWAP(qbits[0], qbits[4]))  #swaping qbits 0 and 4
# print(circuit)

# circuit.append(cirq.rz())

rot = cirq.XPowGate(global_shift=0, exponent=0.5) #rotate by pi/2,  e^(i*pi/2)␣
 ↪phase shift
circuit.append(rot(qbits[0]))
print(circuit)
```

```
[cirq.GridQubit(0, 0), cirq.GridQubit(0, 1), cirq.GridQubit(0, 2),
cirq.GridQubit(1, 0), cirq.GridQubit(1, 1), cirq.GridQubit(1, 2),
cirq.GridQubit(2, 0), cirq.GridQubit(2, 1), cirq.GridQubit(2, 2)]
(0, 0):  H  @       ×  X^0.5

(0, 1):  H  X @

(0, 2):  H    X @

(1, 0):  H      X @

(1, 1):  H        X ×
```

[ ]:

```
```