

# Games for today's quantum computers



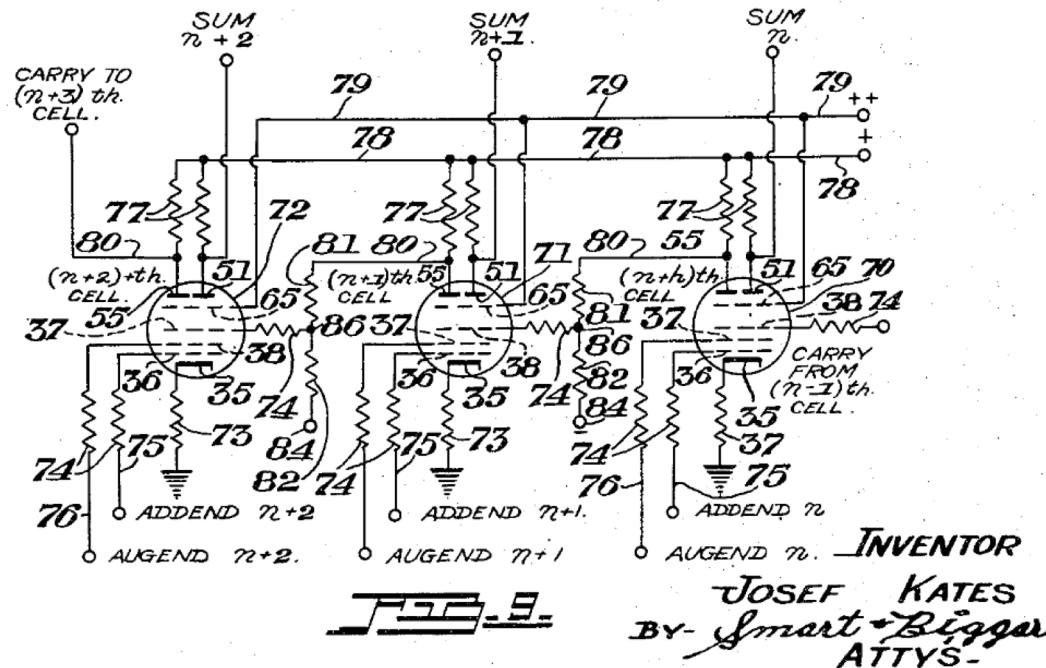
James R. Wootton  
IBM Research - Zurich

# Games for yesterday's computers



- Early games were demonstrations, or for research
- Fun was not a consideration

# Bertie the Brain - 1950



- Tic-Tac-Toe with vacuum tubes and light bulbs
- Built to showcase a new vacuum tube design

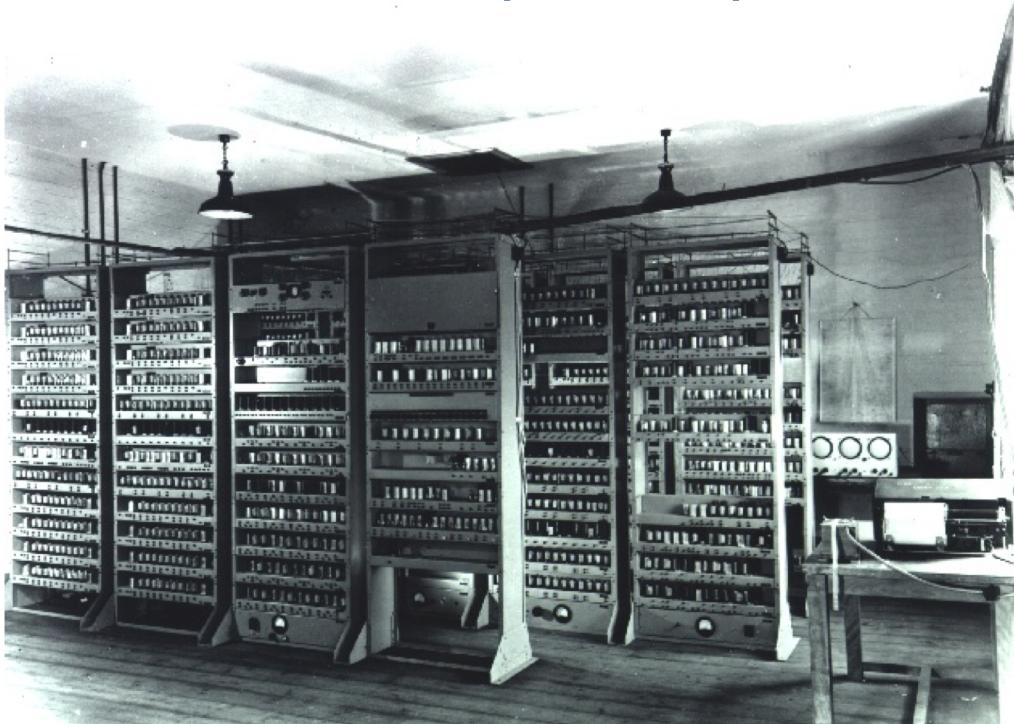
# Nimrod - 1951

“ It may appear that, in trying to make machines play games, we are wasting our time. This is not true as the theory of games is extremely complex and a machine that can play a complex game can also be programmed to carry out very complex practical problems. ”

Pamphlet accompanying the Nimrod sold to Festival of Britain attendees.<sup>[5]</sup>

- Nim with vacuum tubes and light bulbs
- Built to “illustrate the algorithm and programming principles involved” (designer John Bennett)

# Noughts and Crosses (OXO) - 1952



- Tic-Tac-Toe with vacuum tubes and cathode rays
- Built for research into human-computer interaction

# IBM Checkers – 1950s



- Checkers with AI on IBM 700 series
- Built to demonstrate the power of computers  
(and raise IBM's stock price)

Image: [www.ibm.com/ibm/history/ibm100/us/en/icons/ibm700series/impacts](http://www.ibm.com/ibm/history/ibm100/us/en/icons/ibm700series/impacts)

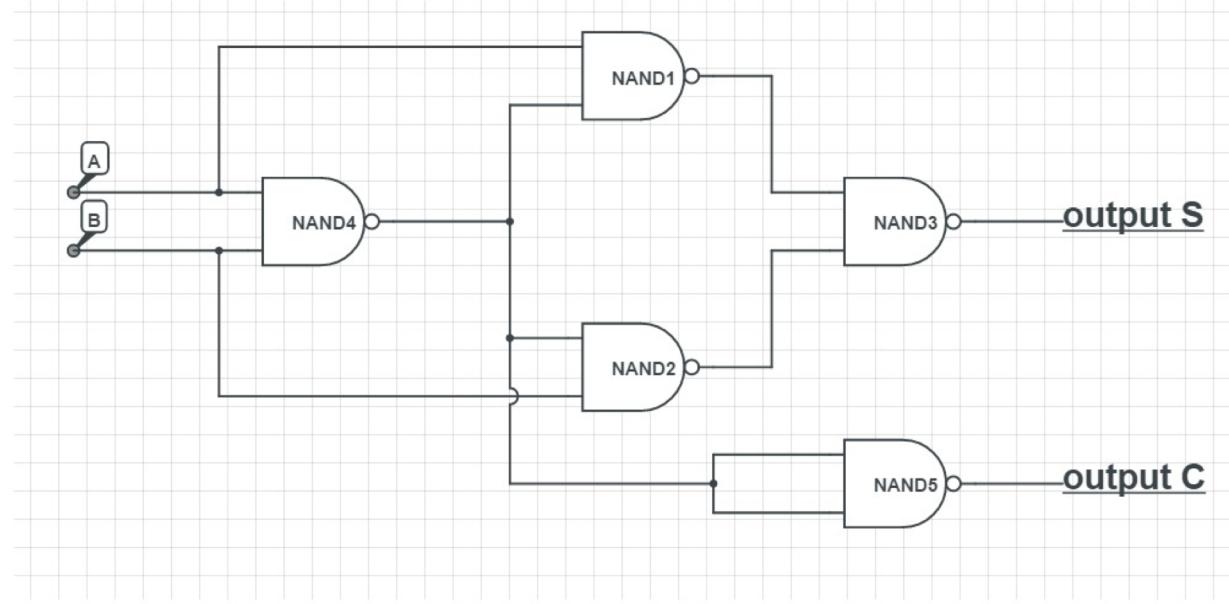
# Spacewar! - 1962



- First novel computer game, running on PDP-1
- Built to
  - Test out a new device (and later, new installations)
  - Showcase its capabilities
  - Be fun!

Image: flickr.com/photos/35034362831@N01/494431001

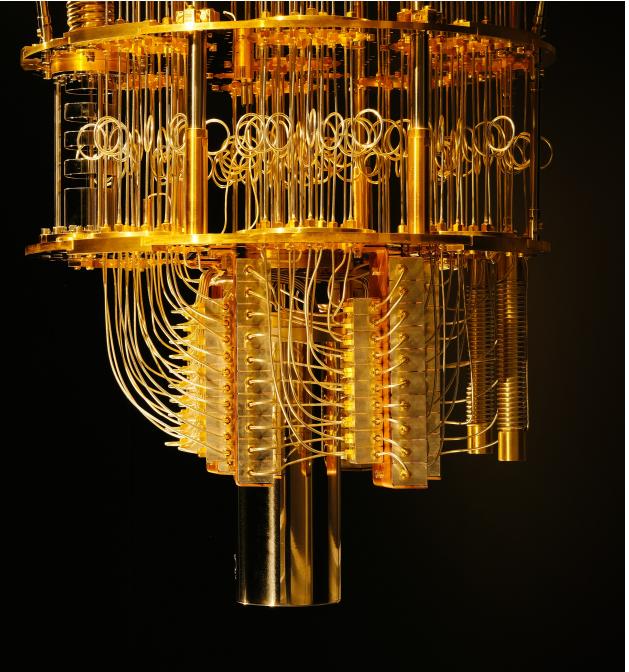
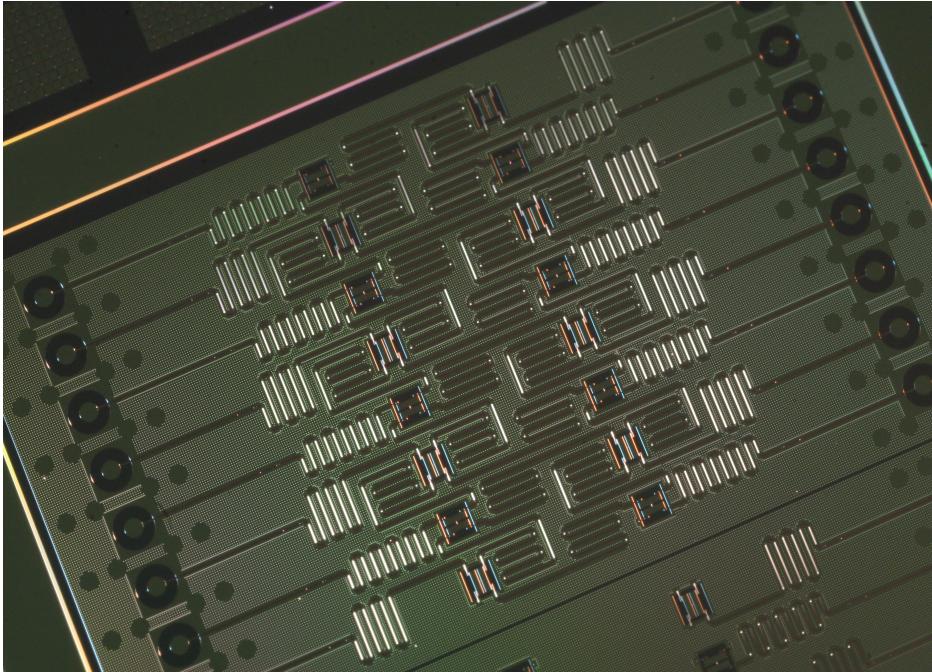
# What are digital computers?



- All information expressed as bits
- All problems solved by comparing pairs of bits
- ‘Computational complexity’ of a problem is the number of these basic operations needed

Image: commons.wikimedia.org/wiki/File:Half\_adder\_using\_NAND\_gates\_only.jpg

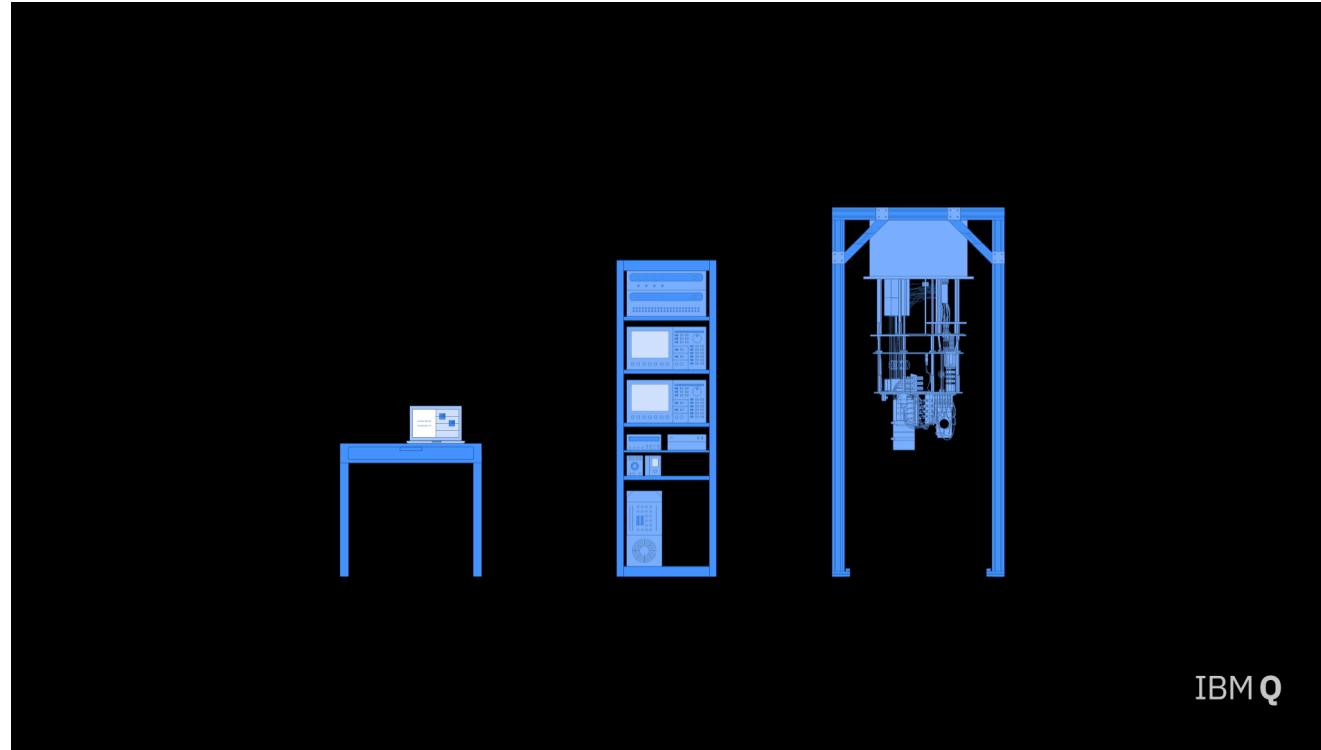
# What are quantum computers?



- A different set of basic operations
- Different hardware to run them on
- Will solve certain problems better than even supercomputers:
  - Better medicine/fuel, more energy efficient logistics, ...

... Image: IBM Research

# Quantum computers are coming out of the lab



- QC has been a topic of science for decades
- Now everyone can access prototype devices
- See ***ibm.biz/start-quantum*** to find out how

# Quantum Battleships - 2017

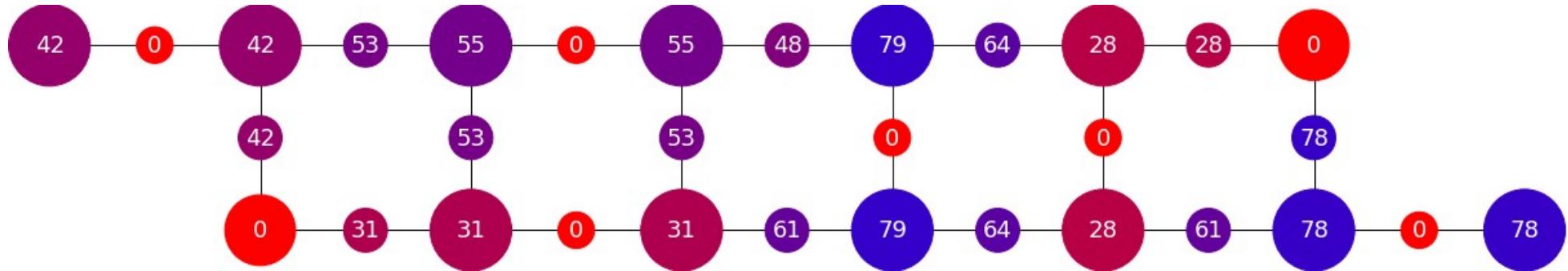
```
q = QuantumRegister(1) # initialize a register with a single qubit
c = ClassicalRegister(1) # initialize a register with a normal bit
qc = QuantumCircuit(q, c) # create an empty quantum program

qc.u3(math.pi,0,0, q[0]) # apply a NOT to the qubit

qc.measure( q[0], c[0] ) # measure the qubit
```

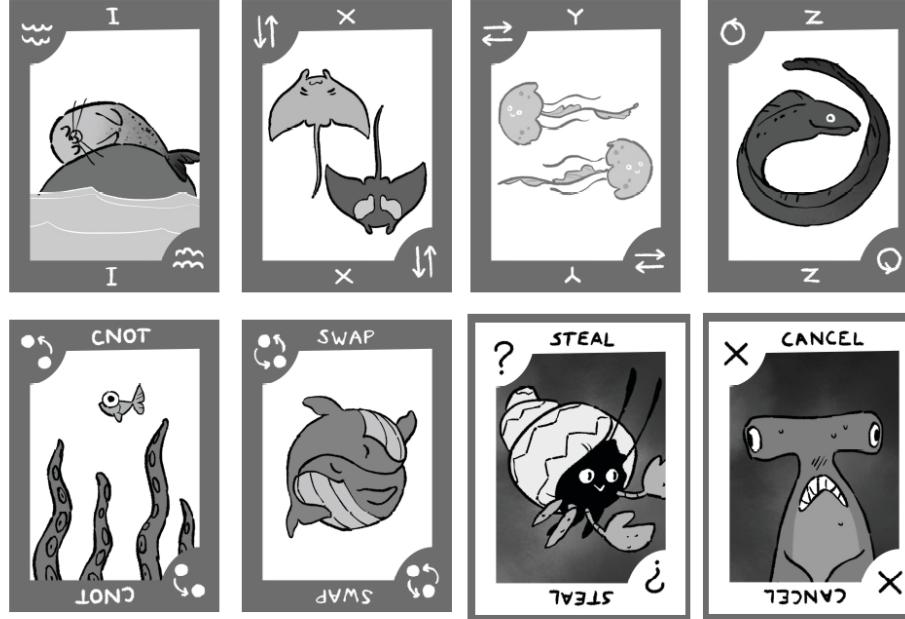
- Japanese-style Battleships built with partial NOT gates
- Forms the heart of a simple quantum programming tutorial
- Games to teach programmers:
  - Implement game mechanic with quantum operations

# Quantum Awesomeness - 2017



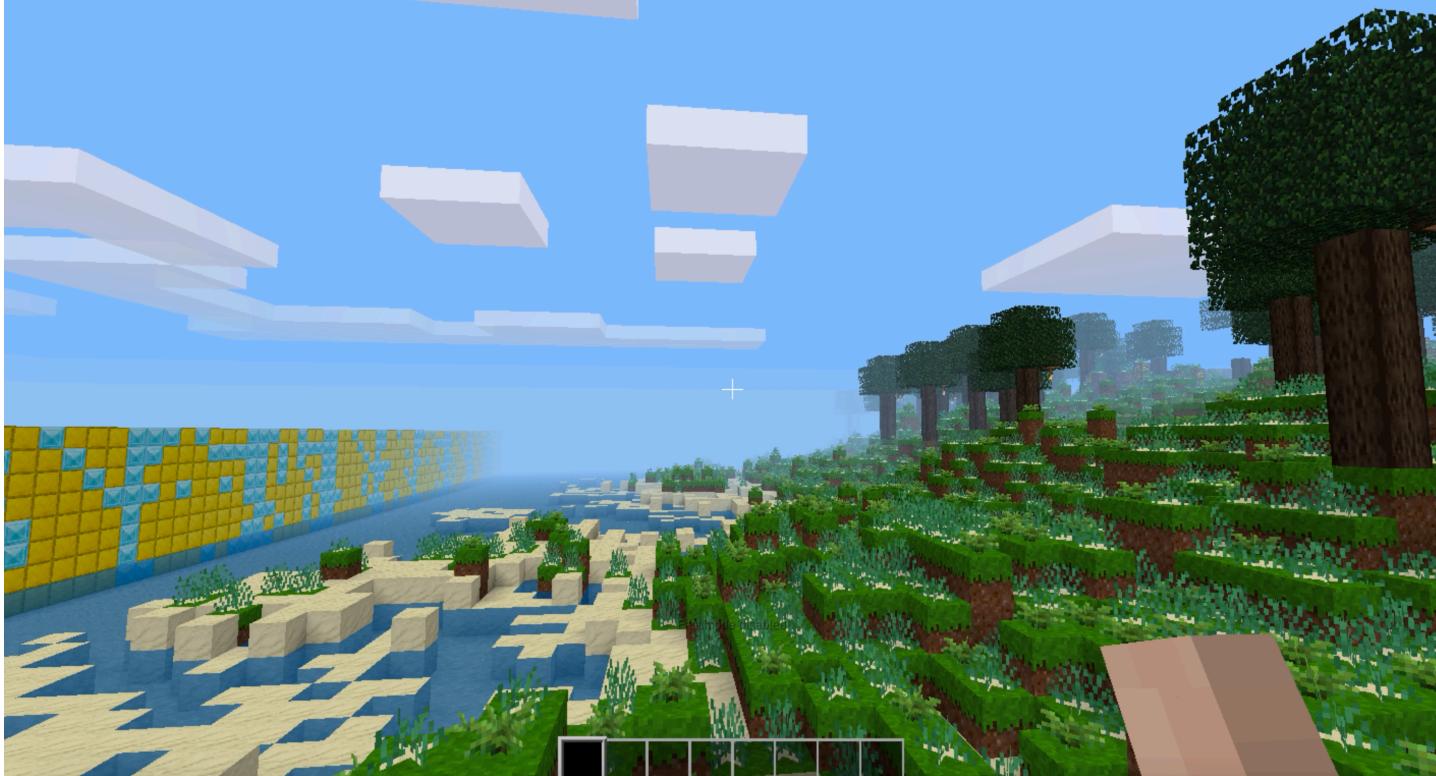
- Simple puzzle game of matching numbers
- Game grid determined by instruction set
- Difficulty determined by noise
- Games to teach players:
  - Gameplay influenced by device specs

# Q|Cards⟩ - 2019



- Card game, where you play quantum gates
- With a real device, noise can be used in strategy
- Games to teach players:
  - Gameplay explicitly based on quantum programming

# Quantum Island Generator



- Quantum effects used to generate texture for map
- Playable game content generated during a loading screen
- Games for fun!

See [ibm.biz/quantumislands](http://ibm.biz/quantumislands)

# + Many using simulators



- Quantum programming inspires and implements game mechanic
- Games run on simulators instead of real devices

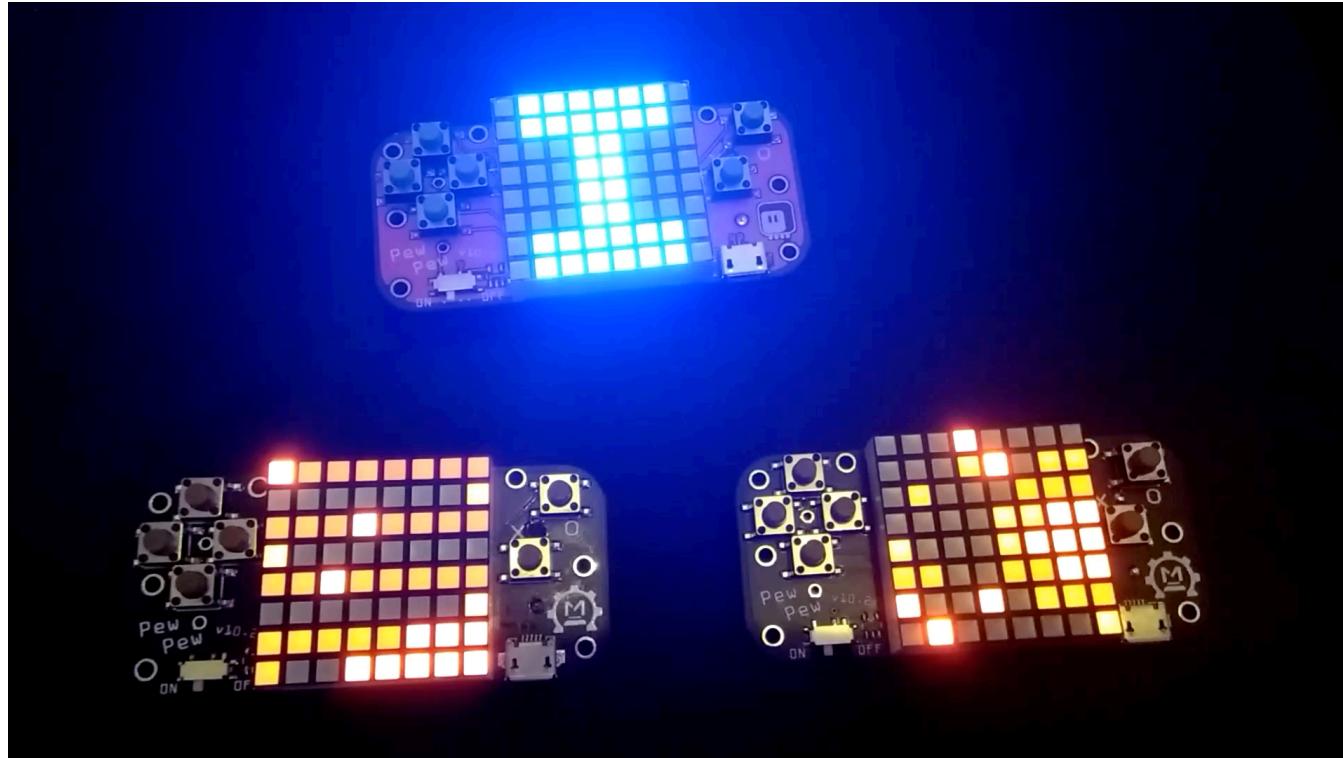
# QiskitBlocks



- A game implemented in Minetest
- Uses simulations of quantum computers
- Build quantum programs
- Learn about quantum as you explore

See [qiskitblocks.org](https://qiskitblocks.org)

# MicroQiskit and PewPews



- A minimal version of Qiskit
- Allows quantum fun on microcontrollers

See [ibm.biz/microqiskit](https://ibm.biz/microqiskit)

# Conclusions



- Games are the perfect way to learn about quantum computing
- Why not make your own?

# Thanks for your attention

- History of games for quantum computers

*ibm.biz/qc-games*

- Quantum island generator

*ibm.biz/quantumislands*

- QiskitBlocks

*qiskitblocks.org*

- MicroQiskit and PewPews

*ibm.biz/microqiskit*

***ibm.biz/start-quantum***

Email [jwo@zurich.ibm.com](mailto:jwo@zurich.ibm.com)  
Twitter @decodoku  
GitHub/Medium @quantumjim

Email  
Twitter  
GitHub/Medium