<u>BUZZ CHRONICLES</u> > <u>GAMING</u> <u>Saved by @Mollyycolllinss</u> See On Twitter

Twitter Thread by Alan Zucconi

Alan Zucconi @AlanZucconi



Fifty years have passed since CONWAY'S GAME OF LIFE firstly appeared on a column called "Mathematical Games" on <u>@sciam.</u>

<u>While</u> most Programmers & Computer Science enthusiasts are familiar with it, not many know that the game is actually TURING COMPLETE.

Let's see why.



The quickest way to prove that a system is TURING COMPLETE is to show that it allows for the constructions of LOGIC GATES. ■■

So, let's see how the ■■■, ■■ and ■■■ gates can actually be constructed in Conway's Game of Life...

Firstly, we need to find a way to encode binary signals.

One very popular choice is to use a stream of GLIDERS. The so-called GOSPER GLIDER GUN can generated a new glider every 30 generations. ■

Hence, receiving a glider every 30 generations counts as a "1".



When two GLIDERS hit each other in just the right way, they both get destroyed.

This means that a GLIDER GUN can stop an incoming glider stream!

We can exploit this mechanism to simulate a NOT gate:



With the same principle, and AND gate can be also constructed by extending a NOT gate.

■■ 0 ■■■ 1 = 0 ■■ 1 ■■■ 0 = 0



For the glider stream to the right to survive an AND gate, the second input needs to cancel the stream coming from the glider gun to the right.

■■ 1 ■■■ 1 = 1



Another small modification allows to construct an OR gate.

■■ 0 ■■ 0 = 0



■■ 0 ■■ 1 = 1 ■■ 1 ■■ 1 = 1



The AND, OR & NOT gates are said to be FUNCTIONALLY COMPLETE, as can be used to construct any logic expression. This is one step away from TURING COMPLETENESS. ■

What we need is a memory block! The pattern below works as a SET-RESET LATCH: a simple 1-bit memory register!



LOGIC GATES & LATCHES are everything needed to build an actual computer.

If you are interested to learn more about this, this short documentary goes into great length to explain the process of building an actual computer in Conway's Game of Life.

https://t.co/7e3LKmGfNi

If you enjoyed this thread, follow me for more content like this!

I tweet about Programming, Artificial Intelligence, Game Development & Shader Coding! ■