Digital Logic and Minecraft Implementation

Author: Jian Wei Heng, Xu Shun Jie, Lu Yong Han

Fu Jen Catholic University

May 21, 2025

Table of Contents

1. Digital Logic Basics
2. Minecraft Implementation

Basic Concepts of Digital Logic

- Digital logic uses binary system (0 and 1)
- 0 represents low voltage (Low)
- 1 represents high voltage (High)
- These states can be implemented through circuits

Figure: Example of 0 and 1 in Minecraft Redstone Circuit

Basic Logic Gates

- AND Gate: Output is 1 only when both inputs are 1
- OR Gate: Output is 1 when any input is 1
- NOT Gate: Output is opposite of input
- NAND Gate: AND Gate output inverted
- NOR Gate: OR Gate output inverted

AND Gate in Minecraft

- The AND gate outputs 1 only when both inputs are 1.
- In Minecraft, you can use redstone components to build an AND gate.
- Below are examples of different input combinations and their outputs:

Input: 0, 0

Input: 0, 1

Input: 1, 1

Figure: AND Gate in Minecraft: Different Input Combinations

Logic Gate Truth Tables

AND Gate

Input A	Input B	Output
0	0	0
0	1	0
1	0	0
1	1	1

Logic Gate Truth Tables

OR Gate

Input A	Input B	Output
0	0	0
0	1	1
1	0	1
1	1	1

Logic Gate Truth Tables

NOT Gate

Input	Output
0	1
1	0

Redstone System in Minecraft

- Redstone Dust: Transmits signals
- Redstone Torch: Generates signals
- Redstone Repeater: Delays and amplifies signals
- Redstone Comparator: Compares signal strengths

Logic Gate Implementation in Minecraft

Basic Logic Gates

- NOT Gate: Using redstone torch
- AND Gate: Using two redstone torches in series
- OR Gate: Using two redstone torches in parallel

Note

In Minecraft, the on/off state of redstone torches is opposite to the logic gate input/output

Practical Applications in Minecraft

- Automatic Door Systems
- Trap Mechanisms
- Automatic Farms
- Item Sorting Systems

Thank You

Thank you for listening!