PIPIRIGEANU VALENTINA

GROUP 1232B

**8X8 RAM**

**ABSTRACT**

This report documents the process of designing and implementing an 8x8 Random Access Memory (RAM) using Logisim, a digital logic simulator. The project explores four distinct configurations: the internal structure of a 8x8 RAM, an 8x8 RAM built from 4x4 RAM modules, an 8x8 RAM with outputs to a teletype (TTY) using a random generator and counter, and an 8x8 RAM using a counter with keyboard input and TTY output. The aim is to demonstrate the principles of digital memory design and operation within a simulated environment.

**INTRODUCTION**

Random Access Memory (RAM) is a fundamental component within computers and other digital devices, serving as the primary workspace for processing data. It allows the system's processor to access any part of the data as quickly as any other part, hence the term "random access." RAM is volatile, meaning it only retains data while powered on, making it ideal for tasks that require fast, temporary data storage. Understanding RAM's function is essential for both computer users and designers, as it directly impacts a device's speed and capability to run programs. In educational contexts, simulating RAM, such as through Logisim, provides invaluable insight into the intricacies of memory operation and system performance.

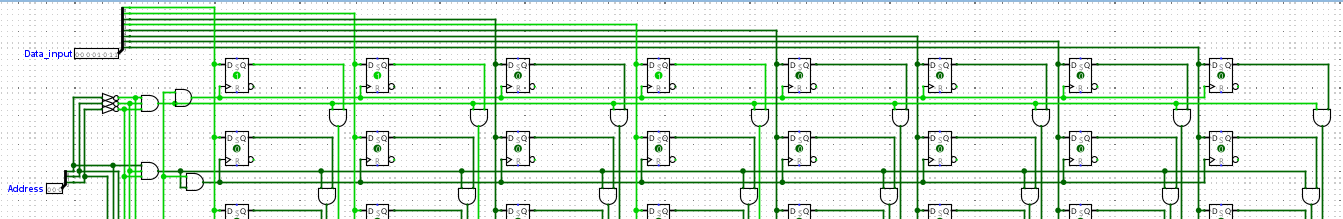
**INTERNAL STRUCTURE OF 8X8 RAM**

A green lines on a white background

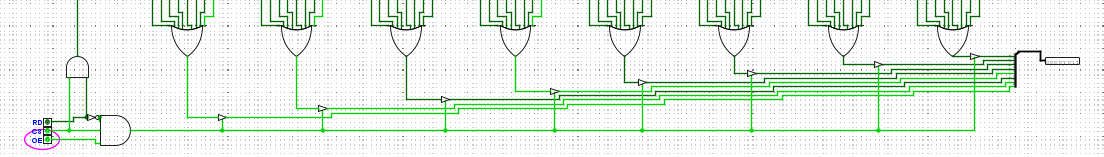
Description automatically generated

The memory is made up of 64 D flip-flops arranged in an 8x8 grid. Each flip-flop stores a single bit of information. An 8-bit data input line allows data to be fed into the RAM. Corresponding 8-bit data output lines are connected to the outputs of the flip-flops, from where the data can be retrieved. The address lines select the specific memory location within the 8x8 matrix. These lines are decoded to target a particular row for either reading or writing operations.

**Data Reading**: To read data from the RAM, the OE line must be inactive (low), and the RD line must be active (high). With these signals, the data stored in the flip-flops is presented on the output lines. The address lines again specify which set of flip-flops is accessed to output the stored data.

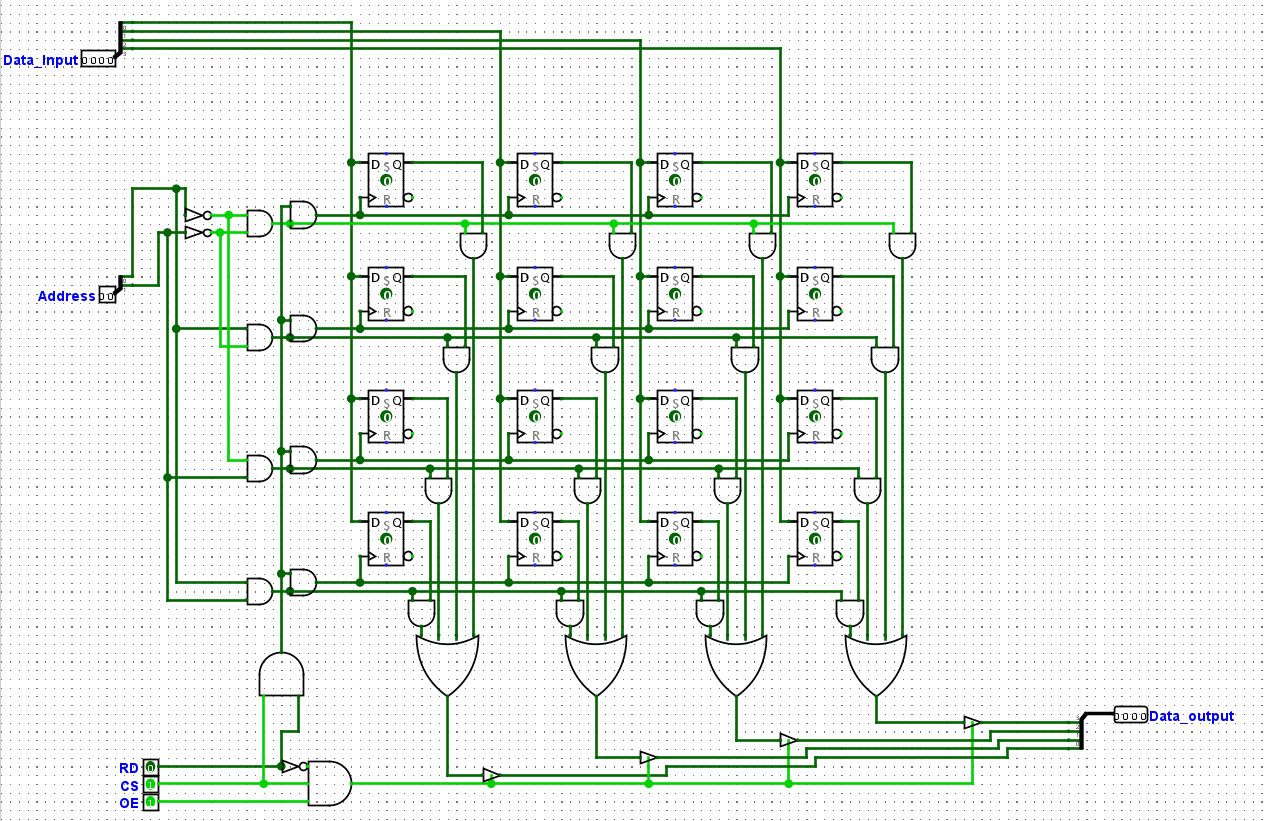


**Data Writing**: To write data into the RAM, the OE line must be active (high), and the RD line must be inactive (low). This setup allows the input data to be stored in the selected memory cells. The addressing lines dictate which specific set of flip-flops (corresponding to an 8-bit word) will receive and store the input data.



**8X8 RAM USING 4X4 RAM MODULES**

First, a 4x4 RAM module circuit was made in the same way as the internal structure of 8x8 RAM.



**RAM Modules**: There are four 4x4 RAM modules (RAM1, RAM2, RAM3, RAM4), each capable of storing 16 bits of data. Together, they form an 8x8 RAM, representing a total of 64 bits of storage.

**Data Input**: The 8-bit data input is split into two groups

* The first 4 bits (lower nibble) are fed into RAM1 and RAM3.
* The last 4 bits (upper nibble) are fed into RAM2 and RAM4.

**Addressing and Control:**

The address lines (presumably A0 and A1) are connected to all RAM modules to select one of the four addresses within each module.The third address line (A2) determines which pair of RAM modules is active—when A2 is active (high), RAM3 and RAM4 (second row) are enabled; when A2 is inactive (low), RAM1 and RAM2 (first row) are enabled.

**Read and Write Operations**

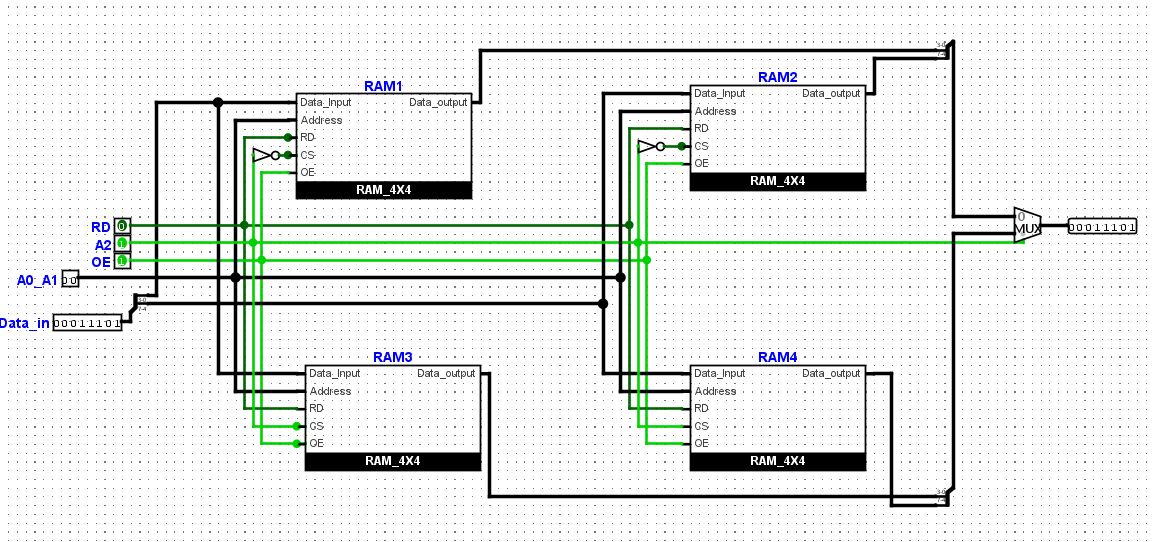
The read (RD), chip select (A2), and output enable (OE) signals are used to control the read and write operations.Depending on the state of A2, either the first row (RAM1 and RAM2) or the second row (RAM3 and RAM4) will perform the read or write operation.

**Data Output Selection**:

The outputs of RAM1 and RAM2 are combined to form one 8-bit output, and the outputs of RAM3 and RAM4 form another 8-bit output.A multiplexer (MUX) is used to select which 8-bit output is active based on the state of A2. If A2 is active, the output from RAM3 and RAM4 is selected; if A2 is inactive, the output from RAM1 and RAM2 is chosen.

**Overall Functionality**:

The circuit effectively creates an 8x8 RAM by using the 4x4 modules in a way that expands both the bit width and the number of addressable locations.A single 8-bit output is provided, determined by the most significant bit of the address (A2), ensuring that at any given time, data from only one set of RAM modules is presented at the output.



**8x8 RAM with TTY Output Using Random Generator and Counter**

**Random Generator**: This component generates random 8-bit values that are used as the data input for the RAM. It simulates dynamic data that might be written to the RAM during typical operation.

**Counter**: The counter cycles through all possible address locations in the 8x8 RAM. It likely increments with each clock cycle and provides the current address to the RAM's address inputs. This ensures that data from the random generator is written to each address sequentially.

**RAM Module**: The 8x8 RAM stores the data. Each address location in the RAM can be accessed individually by the address provided by the counter.

**Control Signals**:The RD (Read) signal, when active, enables the reading of data from the RAM to the output.The CS (Chip Select) signal enables the RAM, allowing it to respond to read and write requests.The OE (Output Enable) signal, when active, allows the output from the RAM to be sent to the connected display (TTY).

**Display (TTY):** The TTY display is used to visualize the data stored in the RAM. It shows the output data corresponding to the address currently provided by the counter.

In operation, the counter would run continuously, cycling through each address in the RAM. As it changes, the random generator provides a new data value, which is written to the RAM at the current address. The TTY display then shows the data being read from these addresses, creating a visual demonstration of writing and reading operations in the RAM.

**A computer screen shot of a computer

Description automatically generated**

**8x8 RAM with Keyboard Input and TTY Output**

**Keyboard**: This allows a user to input 8-bit data into the RAM.

**Counter**: It sequentially generates addresses for the RAM, determining which memory cell the data from the keyboard is written to or read from.

**8x8 RAM**: This stores the data from the keyboard at the location indicated by the counter. The RAM responds to control signals for reading (RD) and chip enable (CS).

**Display (TTY):** Visualizes the data from the RAM's current address as indicated by the counter, showing the operation of the memory in real-time.

Data is entered through the keyboard and written to the RAM at an address specified by the counterThe counter advances sequentially, guiding which memory cell is accessed for each operation.The RAM stores the input data and allows for it to be read based on the counter's address. A TTY display shows the stored data as the counter progresses through the RAM addresses, providing a visual output.

A computer screen shot of a computer

Description automatically generated

**Results and Discussion**

Each setup worked as planned. There were some issues with getting the timing of the signals right and making sure data didn't get mixed up, especially when changing from reading to writing data. The tests showed that it's really important to use control signals in the right order to make sure data is picked up correctly.

**References**

* For internal structures of 8x8 RAM, 4X4 RAM and 8x8 RAM contructed from 4x4 RAM :

<https://www.asmarya.edu.ly/journal2/wp-content/uploads/2017/06/27_5.pdf>

* For 8x8 RAM with Random Generator and for 8X8 RAM with Keyboard:

https://www.youtube.com/watch?app=desktop&v=lk5iSGwzmgY&ab\_channel=NeutronNick11