Activity 7: Memory
Recorder's Report

Manager:

Reader:

Recorder:

Driver:

Date:

Score: Satisfactory / Not Satisfactory

Record your team's answers to the key questions (marked with 🔑) below.

a) Model 1, Question #10

b) Model 2, Question #14

c) Model 3, Question #21

Activity 7: Memory

In previous activities, we explored combinational logic circuits where outputs depend only on current inputs. However, computers need to remember information—they need memory. This activity introduces sequential logic circuits, where outputs depend on both current inputs and previous states. We'll build up from the simplest memory element (a flip-flop that stores one bit) to more complex circuits that can store and retrieve multiple bits using addresses. Understanding how memory works at the hardware level is fundamental to grasping how computers store programs, data, and maintain state during execution.

Content Learning Objectives

After completing this activity, students should be able to:

- Students will be able to explain how a flip-flop maintains state using feedback.
- Students will be able to distinguish between level-triggered and edge-triggered D flip-flops.
- Students will be able to describe how addressable memory uses select lines to read or write specific memory locations.
- Students will be able to trace the flow of data through memory circuits.

Process Skill Goals

During the activity, students should make progress toward:

- Analyzing circuit diagrams to understand data storage and retrieval.
- Comparing different types of flip-flops and their timing characteristics.
- Collaborating to trace signals through sequential circuits and build understanding collectively.

Sources

- https://www.codehiddenlanguage.com/

Model 1 Feedback

Recall that a relay consists of a coil of wire around a ferromagnetic material (such as iron) with a switch that is pulled toward the coil when electricity flows because the coil becomes magnetic. The switch connects an input signal or wire to one of two output terminals depending on whether electricity is flowing.

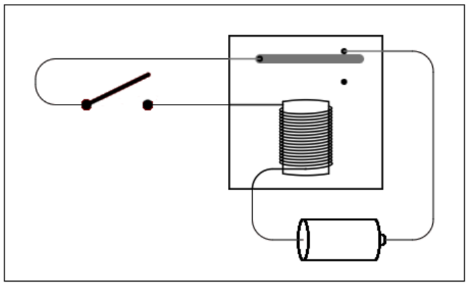

Figure 1: Oscillating Relay

Refer to Model 1 above as your team develops consensus answers to the questions below.

Questions (10 min) Start time:

1. When the switch is open, does electricity flow through the circuit?

2. When the switch is initially closed, does electricity start to flow through the circuit?

3. What happens to the coil on the relay soon after step 2?

4. What happens to the bar on the relay soon after step 3?

5. What happens to the circuit after step 4?

6. What happens to the coil soon after step 5?

7. What happens to the bar soon after step 6?

8. What happens to the circuit after step 7?

9. Summarize what happens when the switch is closed (steps 2-8).

10. Recall that an inverter switches a 0 to 1 and a 1 to 0. What is the output of the following circuit?

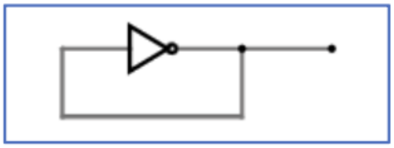

Figure 2: Inverter Feedback

Feedback occurs when the output of a circuit is "fed back" into the input of the circuit. You have heard what happens when the sound from a PA system's speaker gets into a microphone!

Model 2 Flip-Flop

Recall that the output from a NOR gate is 1 only when both inputs are 0. If either input is 1 then the output is 0.

Refer to Model 2 above as your team develops consensus answers to the questions below.

Questions (10 min) Start time:

11. Complete Figure 3 by showing both switches open and highlight (in red if you can) any portion of the circuit with high voltage (assuming both inputs to the left NOR are both 0).

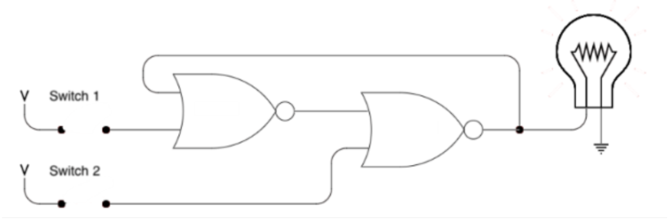

Figure 3

12. With the circuit in the state shown in Figure 3, imagine that Switch 1 is closed. Complete Figure 4 highlighting any portion of the circuit with high voltage.

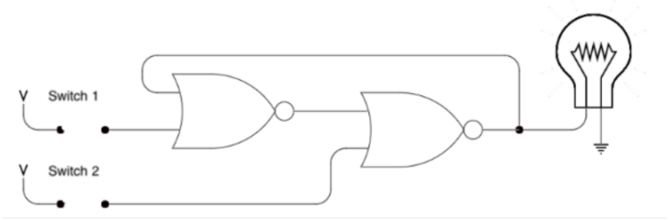

Figure 4

13. With the circuit in the state shown in Figure 4, imagine that Switch 1 is opened. Complete Figure 5 highlighting any portion of the circuit with high voltage.

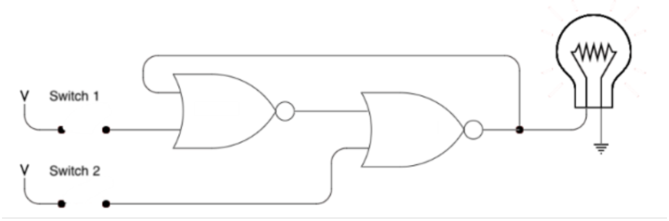

Figure 5

14. Both Figures 3 and 5 have both switches open. How are they different?

15. With the circuit in the state shown in Figure 5 (Switch 1 is open), imagine that Switch 2 is closed. Complete Figure 6 highlighting any portion of the circuit with high voltage.

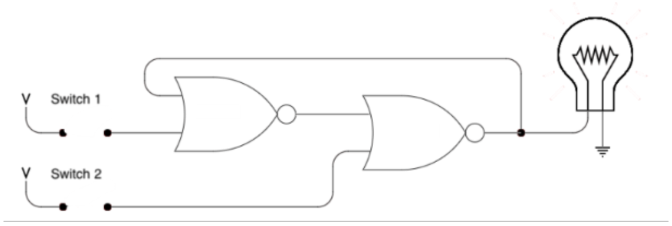

Figure 6

16. With the circuit in the state shown in Figure 6, imagine that Switch 2 is opened. Complete Figure 7 highlighting any portion of the circuit with high voltage.

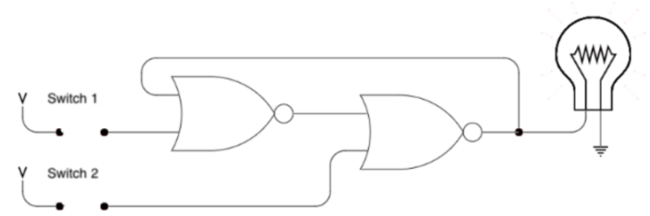

Figure 7

Model 3 Level-Trigger D-Type Flip-Flop

Questions (10 min) Start time:

17. Recall that an AND gate has an output of 1 only if both inputs are 1. In Figure 8 enter the value 0 for Write, Data, and output (Q). Highlight the lines that have high voltage for this state to exist.

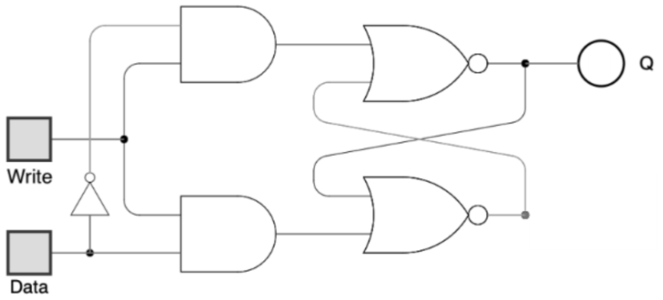

Figure 8

18. With the circuit in the state shown in Figure 8, imagine that Data changes from 0 to 1. Highlight the lines in Figure 9 that have high voltage and update Q.

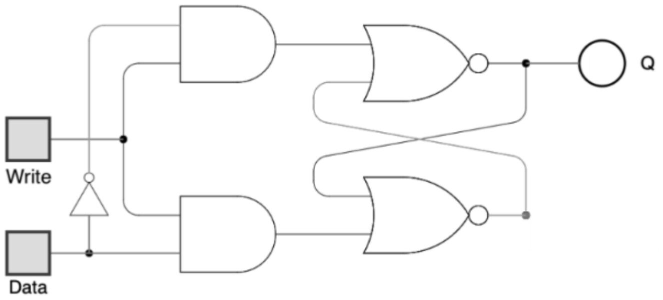

Figure 9

19. With the circuit in the state shown in Figure 10, imagine that Write changes from 0 to 1. Highlight the lines in Figure 10 that have high voltage and update Q.

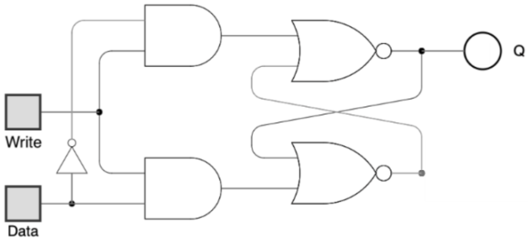

Figure 10

20. With the circuit in the state shown in Figure 10, imagine that Write changes from 1 to 0. Highlight the lines in Figure 11 that have high voltage and update Q.

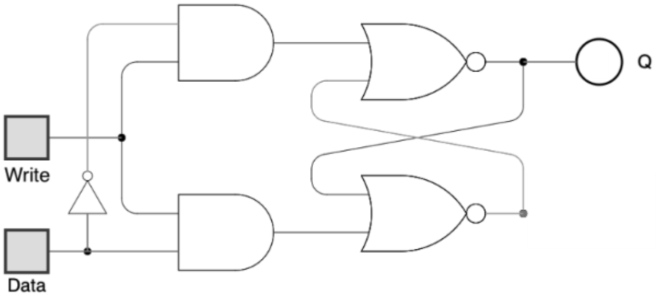

Figure 11

We now have a circuit that latches onto (remembers) a data line only when the write line is 1.

Model 4 Edge-Triggered D-Type Flip-Flop

Questions (10 min) Start time:

21. What simple change would be required to have the flip-flop in Model C remember the data value when the write line is 0 (instead of 1)?

22. The following figure shows two level-trigger D-type flip-flops, along with two inputs (write and data), and one output. Connect the inputs so that the first flip-flop latches the data value when the write line is 0 (the switch is open). Connect the second flip-flop so that it latches the output of the first flip-flop when the write line is 1. Connect the output of the second flip-flop to the LED on the right.

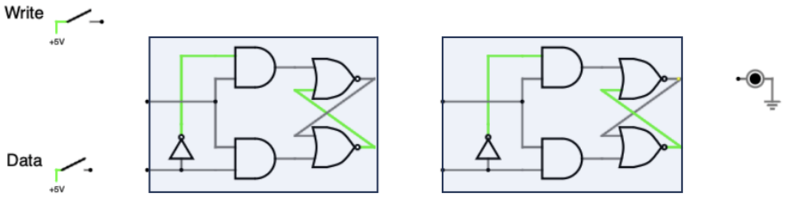

Figure 12

This circuit is an edge-triggered D-type flip-flop that latches a value only when the write line has a positive transition from 0 to 1. Note that Figure 12 is slightly different from that shown on page 229 of the text, but the behavior is the same (which do you like better and why?).

Model 5 Addressable Memory

The latch from Model D has two inputs for each bit, a write line and a data line. If you were going to build many such latches (think millions or billions), it would be nice to have a way to let them share the write and data lines and do the selection with fewer lines. For example, a thousand separate locations could be identified with only 10 lines and a billion locations could be identified with 30 lines. Think of each location as having a unique integer address.

Refer to Model 5 above as your team develops consensus answers to the questions below.

Questions (10 min) Start time:

23. Consider the following 3-to-8 decoder. It has eight latches below (not shown), but only a single data in line and write line. A three-bit address is added on the left.

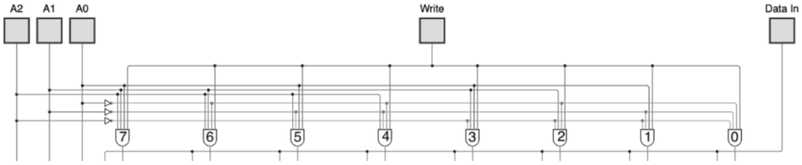

Figure 13

a) If the write line is 1 and each of the address lines (A0, A1, and A2) are 0, what are the output values for the 4-input AND gates labeled 7 to 0?

7	6	5	4	3	2	1	0

b) If the write line is 1 and each of the address lines are 1, what are the values for the 4-input AND gates labeled 7 to 0?

7	6	5	4	3	2	1	0

c) Generalize the relationship between the address lines and the AND gates.

24. Just like it would be nice to have a single data-in line for many latches, it would be nice to have a single data-out line. An 8-input OR gate would have a 1 output if any of the latches had a 1 output. How could you modify Figure 13 to select which latch to read? (After thinking about it, see Figure 14 for a hint!)

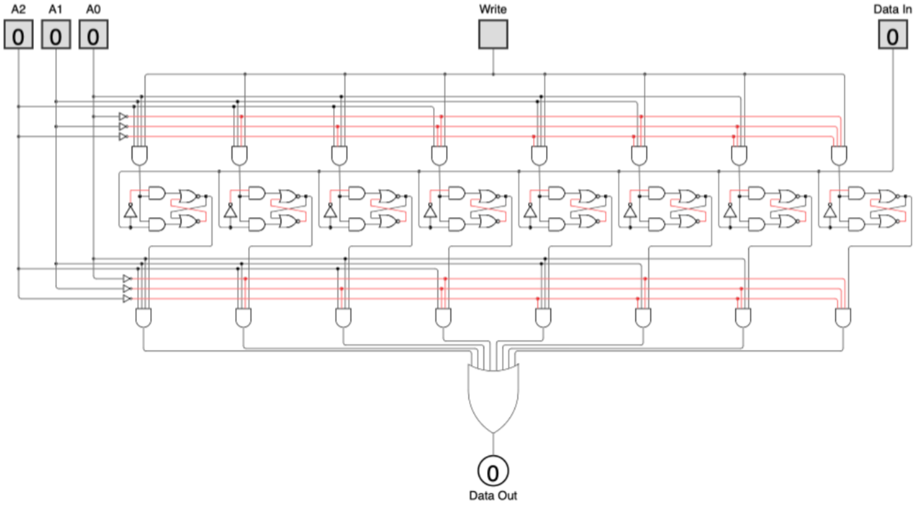

Figure 14

Figure 14 shows an addressable 8-bit array of memory. The address control lines specify which bit to read or write. There is a single data-in line that can be used to store to memory (if the write control line is enabled) and read from memory. Instead of storing eight bits these circuits could be stacked eight high to store eight bytes. The address and write control lines would be shared, and the data lines would be unique for each bit of the byte (so 20 lines total). Instead of having only eight locations (with a three-bit address), these circuits could be expanded to have (say) a 16-bit address and 65,536 locations (64 KiB of RAM). We now have addressable memory!