Combinational and sequential logic are two fundamental types of logic circuits. They differ in how they process and store information.

Combinational logic circuits produce an output that is a direct function of its inputs. This means that the output of a combinational logic circuit depends only on the current values of its inputs. Combinational logic circuits do not have any memory, so they cannot store information from one cycle to the next.

Sequential logic circuits, on the other hand, produce an output that is a function of its inputs and its current state. This means that the output of a sequential logic circuit depends not only on the current values of its inputs, but also on the values of its inputs in previous cycles. Sequential logic circuits have memory, so they can store information from one cycle to the next.

Here is a table that summarizes the key differences between combinational and sequential logic:

Feature	Combinational logic	Sequential logic
Output	Direct function of inputs	Function of inputs and current state
Memory	No	Yes
Examples	Adders, multiplexers, decoders	Flip-flops, counters, registers

Combinational logic circuits are used in a wide variety of applications, including arithmetic circuits, control circuits, and data manipulation circuits. Sequential logic circuits are used in applications where memory is required, such as counters, registers, and shift registers.

Here are some examples of combinational logic circuits:

- **Adders:** Adders are used to add two numbers together.
- **Multiplexers:** Multiplexers are used to select one of several inputs to send to an output.
- **Decoders:** Decoders are used to convert a binary code to a decimal code.

Here are some examples of sequential logic circuits:

- **Flip-flops:** Flip-flops are used to store a single bit of information.
- **Counters:** Counters are used to count the number of occurrences of an event.
- **Registers:** Registers are used to store multiple bits of information.