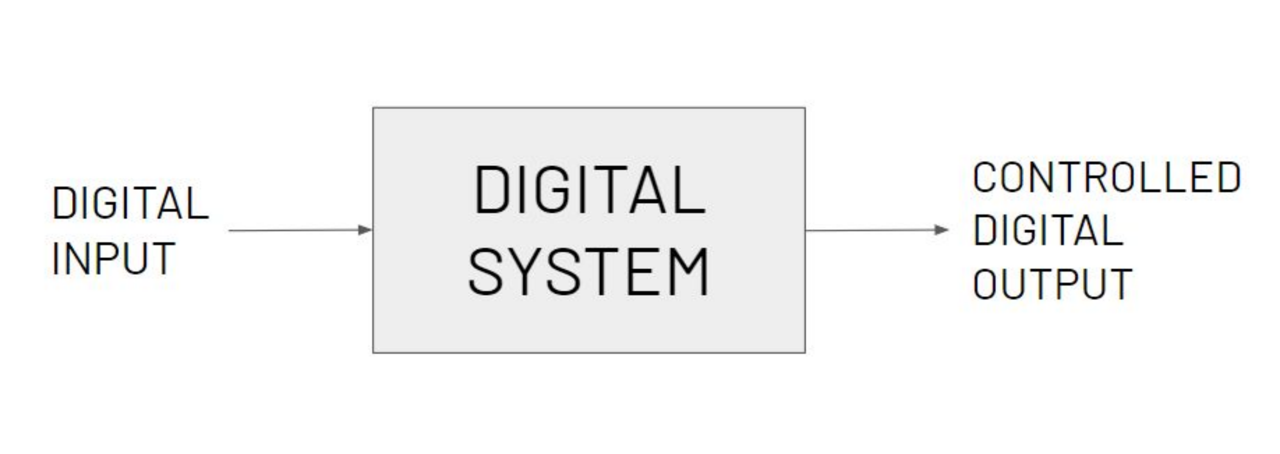
Mohammad Rahimi
11/29/23

Combinational Vs Sequential Logic

Understanding The Fundamentals

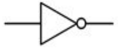
Introduction To Logic In Digital Systems

- Think of digital logic as the backbone of today's computers. It's like a rulebook that helps us solve really complex problems by breaking them down into simpler "yes" or "no" questions.
- Digital logic is like the DNA of modern computers. It's the rulebook that guides how computers tackle tough problems by breaking them into basic "yes" or "no" choices.



Combinational Logic

- In a more relatable way, think of a combinational circuit as an instant decision-maker—it looks at the current information and gives an answer right away, without remembering past data. On the flip side, a sequential circuit is like someone who considers both the current information and what happened before to make a decision—it has a memory to help it choose.
- Examples: Logic gates (AND, OR, NOT), multiplexers, decoders.

Logic Gate	Symbol	Description	Boolean
AND		Output is at logic 1 when, and only when all its inputs are at logic 1,otherwise the output is at logic 0.	$X = A \cdot B$
OR		Output is at logic 1 when one or more are at logic 1.If all inputs are at logic 0,output is at logic 0.	$X = A + B$
NAND		Output is at logic 0 when,and only when all its inputs are at logic 1,otherwise the output is at logic 1	$X = \overline{A \cdot B}$
NOR		Output is at logic 0 when one or more of its inputs are at logic 1.If all the inputs are at logic 0,the output is at logic 1.	$X = \overline{A + B}$
XOR		Output is at logic 1 when one and Only one of its inputs is at logic 1. Otherwise is it logic 0.	$X = A \oplus B$
XNOR		Output is at logic 0 when one and only one of its inputs is at logic1.Otherwise it is logic 1. Similar to XOR but inverted.	$X = \overline{A \oplus B}$
NOT		Output is at logic 0 when its only input is at logic 1, and at logic 1 when its only input is at logic 0.That's why it is called and INVERTER	$X = \overline{A}$

Characteristics Of Combinational Logic

- Instantaneous Output: It means that at any given moment, the sensor's output instantly reflects the input it's getting right then. This type of sensor reacts quickly compared to others that take more time to process and give an output.
- Truth Tables: Think of truth tables as maps for figuring out what's true in an expression. Each column represents something we're looking at, like different parts of a problem, and each row is a different way those parts could be true or false. The table helps us see the end result for each combination of truths or falsities.
-

p	q	$p \wedge q$	These are statements. "$p \wedge q$" stands for "p and q."
T	T	T	The first row says "p" is true, "q" is true, and "$p \wedge q$ is true."
T	F	F	The second row says "p" is true, "q" is false, and "$p \wedge q$" is false.
F	T	F	The third row says "p" is false, "q" is true, and "$p \wedge q$ is false.
F	F	F	The fourth row says "p" is false, "q" is false, and "$p \wedge q$" is false.

Sequential Logic

What is Sequential Logic? Sequential logic is like the memory center in a digital circuit. It's made up of storage and combinational parts. Unlike the circuits we know, it doesn't just react to the input—it also considers its past states. The output it gives depends on both the input at that moment and what it remembers from before, which is stored in its memory.

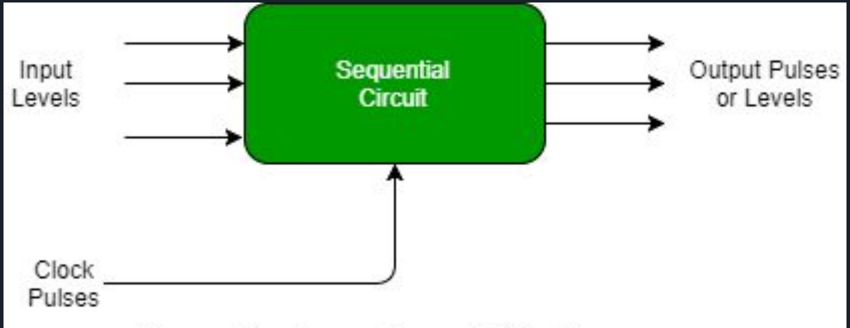
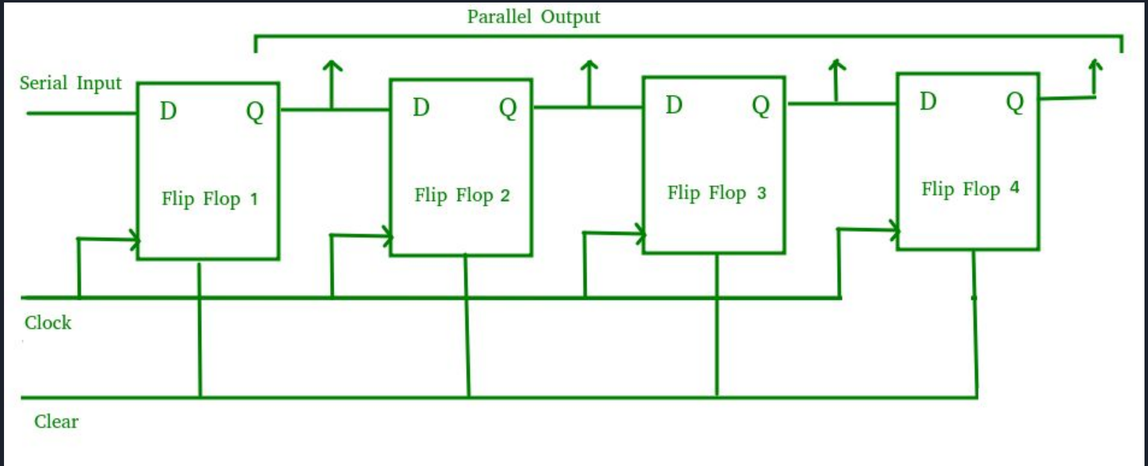
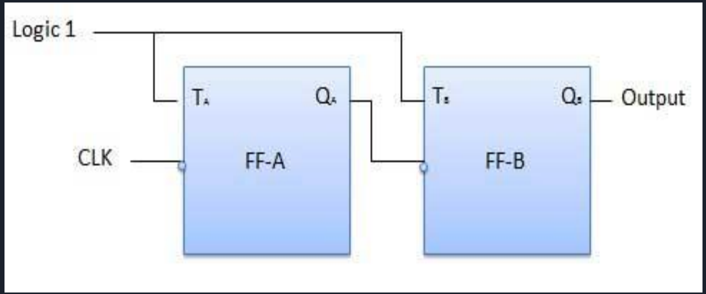


Figure: Synchronous Sequential Circuit

Characteristics Of Sequential Logic

- **Flip-Flops:** A flip flop, in simple terms, is a type of circuit that takes snapshots of its inputs and updates its outputs at specific moments, not constantly.
- **Shift Registers:** A shift register is basically a chain of flip-flops linked together, where what comes out of one flip-flop goes into the next. It all starts with the first flip-flop getting a serial input.
- **Counters:** A counter is like a sequence tracker made of flip-flops. It helps count clock cycles, which happen at regular times. This way, it can measure things like time periods or how often something happens.





Differences Between Combinational and Sequential Logic

- **Timing-** Combinational circuits don't care about time—they only look at what's happening right now. Sequential circuits, though, are like clock-watchers. They take cues from both what's happening now and what happened before to decide what to do next.
- **Memory:** In a combinational circuit, the output only cares about what's happening right now. It doesn't have a memory, so it can't remember anything from before.
- On the flip side, in a sequential circuit, the output looks at both what's happening now and what happened before. It's got a memory element (like a brain) that helps it remember past stuff.
- **Output dependency:** The output of a sequential circuit depends on what's happening in the present and what happened before. On the other hand, a combinational circuit only looks at what's happening right now. It keeps things simple because it focuses solely on the present values of its input.

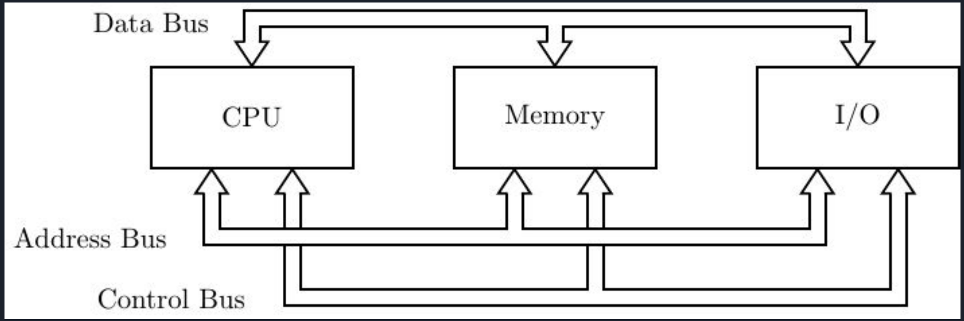
Applications And Use Cases

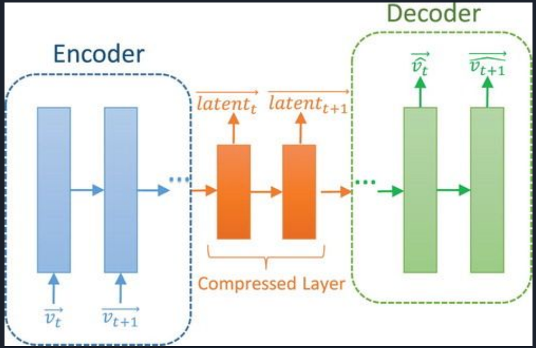
Combinational Circuits:

- Arithmetic Logic Units (ALUs): These circuits handle mathematical operations like addition and subtraction in computers.
- Encoders and Decoders: Encoders convert data into specific formats, while decoders do the opposite by translating signals into understandable forms.

Sequential Circuits:

- Memory Units: They store information, ensuring your computer remembers things even when powered off.
- Processors and State Machines: They're the brains of computers, managing tasks and using memory to keep track of where they are in a process.





Design Considerations

When picking between combinational and sequential logic for a design, here's what to think about:

- **Speed Demands:** If your system needs lightning-fast responses, combinational might be the way to go—it reacts instantly. Sequential takes its time, considering both present and past inputs.
- **Memory Requirements:** Do you need memory to store past information? Sequential circuits have a knack for remembering, while combinational circuits just focus on the here and now.
- **Complexity Concerns:** Combinational circuits are simpler in structure, dealing only with current inputs. Sequential ones, with their memory elements, can get pretty complex.
- Scalability Needs: Thinking about future expansions? Sequential circuits, despite their complexity, might offer more room for growth compared to combinational ones.

Summary and conclusion

- **Combinational vs. Sequential:** Combinational logic is like an instant decision-maker—it reacts immediately to current inputs. Sequential logic, on the other hand, takes its time, considering both now and before to make choices.
- **In Digital Systems**: Both combinational and sequential logic play crucial roles. Combinational circuits process things instantly, while sequential circuits manage memory and past information.
- **Working Together:** They're like a tag team in digital systems. Combinational logic handles the quick tasks, while sequential logic manages the memory-heavy, thoughtful processes.
- Importance in Design: Understanding these concepts is vital in digital design. Knowing when to use instant processing or memory-based decision-making helps create efficient and functional systems. Both are crucial puzzle pieces in designing today's digital world.