1. Finite State Machines (FSMs)

- **Definition**: A model with a finite number of states, transitions triggered by inputs.
- **Used for**: Modeling behavior where the *order of inputs matters* (e.g. string recognition, coin counters, locker combos).
- **Examples**:
    - Detect strings ending in 'a' or 'b'.
    - Accept $5 in $1/$2 coins.
    - Bit sequences ending in 1 but excluding 00.
    - 1-bit storage using FSM logic.

2. FSMs to Circuits

- **Feedback loops** in FSMs = sequential circuits.
- **Sequential Logic**: Stores information over time using feedback (Q feeds back into logic).
- **Problems without refresh**: Signal degrades due to noise/attenuation.
- **Solution**: Use NOT gates for basic refresh or introduce memory elements (like flip-flops).

3. Oscillators and Clock

- **Oscillators**: Circuits that toggle between 0 and 1 based on propagation delay.
- **Clock**: Synchronizes operations in CPU (load → compute → store).
- **Clock Edge Timing**:
    - Positive edge: triggers load or add.
    - Negative edge: triggers store.

---

4. Latches and Flip-Flops

Latches

- **S-R Latch**: Basic 1-bit memory.
    - S (Set) → stores 1.
    - R (Reset) → stores 0.
    - Simultaneous S=R=1 is undefined.

Flip-Flops

- **SR Flip-Flop**: Clocked version of S-R latch (edge-triggered).
- **JK Flip-Flop**: J=K=1 toggles output.
- **D Flip-Flop**: Simplified (D=1 sets to 1, D=0 sets to 0).

5. Registers

- **Definition**: Small, fast storage used directly by CPU.
- **Types**:
    - Program Counter (PC)
    - Memory Address Register (MAR)
    - Memory Buffer Register (MBR)
    - Status Register (flags)
    - General Purpose Registers
- **Loading**:

- **Serial**: Bit-by-bit input.
- **Parallel**: Whole word input (faster).

6. Counters

- Specialized registers that increment automatically.
- Used in program counters and loops.

7. Types of Memory

Type	Volatility	Density	Cost	Use
Magnetic	Permanent	High	Low	HDD
Flash	Permanent	High	Low	SSD
Capacitors	Volatile	Medium	Medium	DRAM (main memory)
Flip-Flops	Volatile	Low	High	SRAM (cache)