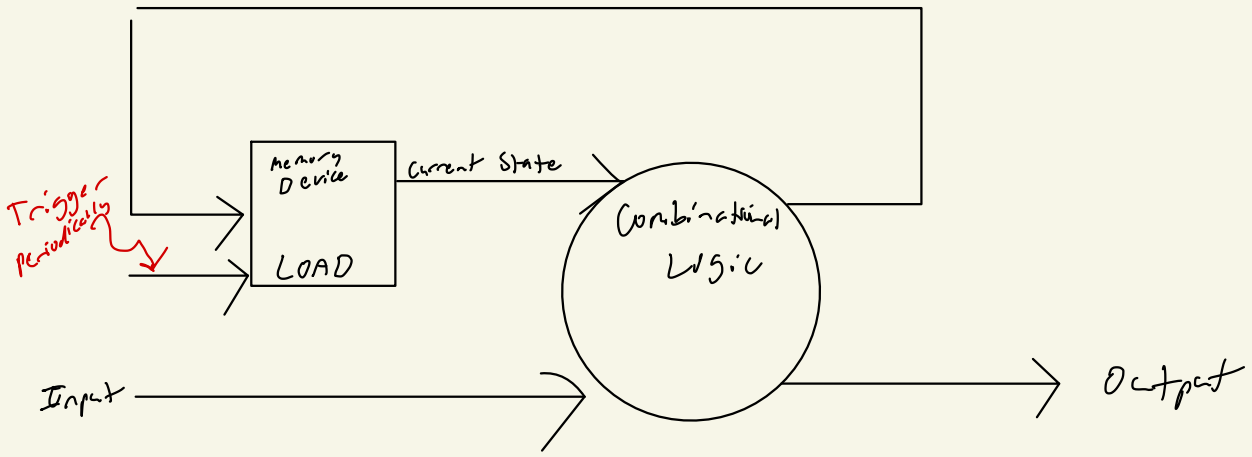
Ch 5 - Sequential Logic

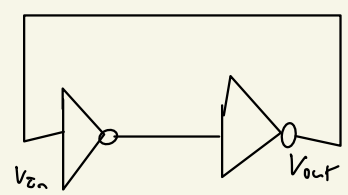
Digital State



- memory stores current state, produced at output
- Combinational logic computes:
  - next state
  - output bits
- State changes on Load control input

Memory : Capacitors vs Feedback

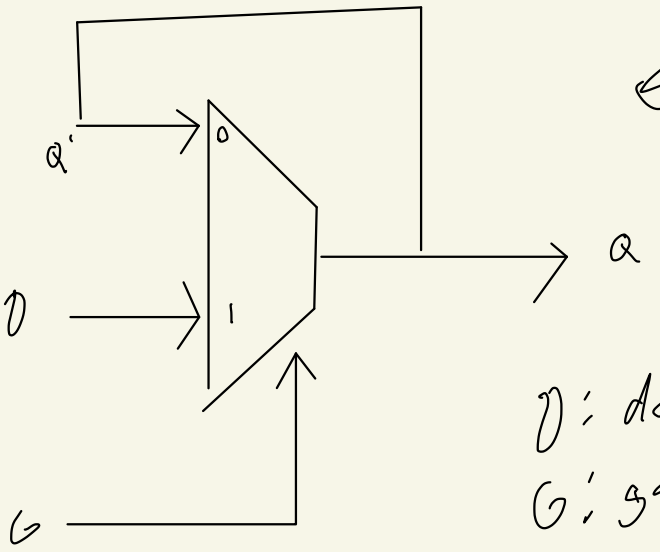
- Capacitor
  - Pros
    - compact / low cost
  - Cons
    - complex interface
    - leaks
  - To write:
    - drive bit line
      - ↓
      - turn on access fet
      - ↓
      - force storage cap to a new voltage
  - To read:
    - precharge bit line
      - ↓
      - turn on access fet
      - ↓
      - detect small change in bit line voltage
- Feedback
  - use positive feedback to store information definitely
    - bistable storage
  - three solutions:
    - two endpoints are stable
    - middle point is metastable



Settable Memory Element

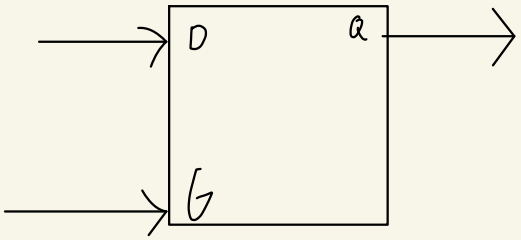
↳ It's easy to build settable storage element (latch) using a level MUX

← D-Latch



D: data input
G: gate input
Q: state output

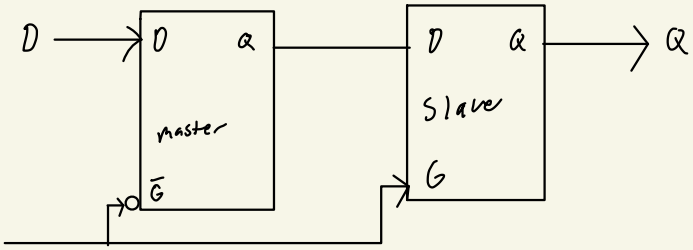
||

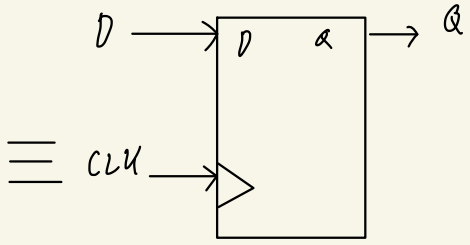


Output valid when:

- G=1, D stable for T_{PD}, independently of Q'
- Q=D stable for T_{PD}, independently of G
- G=0, Q stable for T_{PD}, independently of D

D-Register





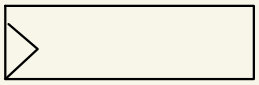
↳ master closed when slave open (vice-versa)

↳ implies no combinational path through register

↳ timing

- ↳ t_{PD} : CLK → Q
- ↳ t_{CD} : CLK → Q
- ↳ t_{SETUP} : guarantees D has propagated through feedback path before master has closed
- ↳ t_{HOLD} : guarantees master is closed and data is stable before allowing D to change

Single-Clock Synchronous Circuits



register → building block

- no combinational cycles
- single periodic clock signal shared among all clocked devices
- only care about value of register data inputs just before rising edge of clock
- period greater than every combinational delay + setup time
- change saved state after noise-inducing logic transitions have stopped