Digital IC Design

Lecture 17
Sequential Circuit Design (1): Latches and Flip-Flops

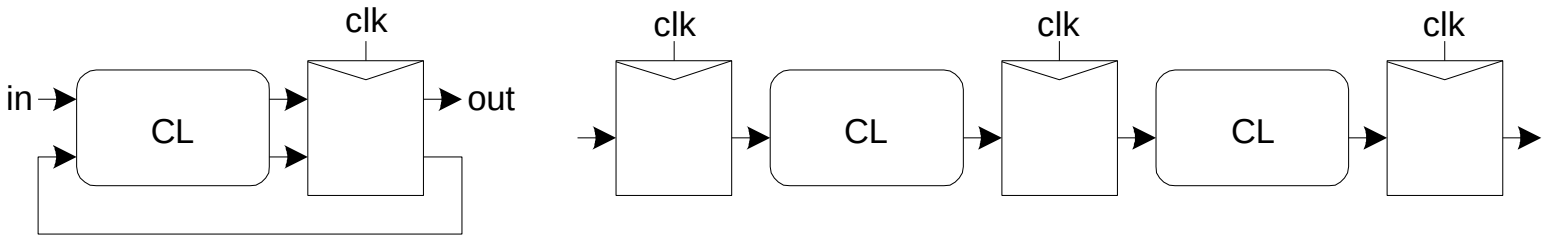
Dr. Hesham A. Omran

Integrated Circuits Laboratory (ICL)
Electronics and Communications Eng. Dept.
Faculty of Engineering
Ain Shams University

This lecture is mainly based on "CMOS VLSI Design", 4th edition, by N. Weste and D. Harris and its accompanying lecture notes

Combinational vs. Sequential Logic

- Combinational logic circuits
  - Output depends on current inputs
- Sequential logic circuits
  - Output depends on current and previous inputs
  - Use flip-flops or latches (memory/sequencing elements)
  - Separate previous, current, and next states (tokens)
  - Ex: FSM, pipeline



Finite State Machine

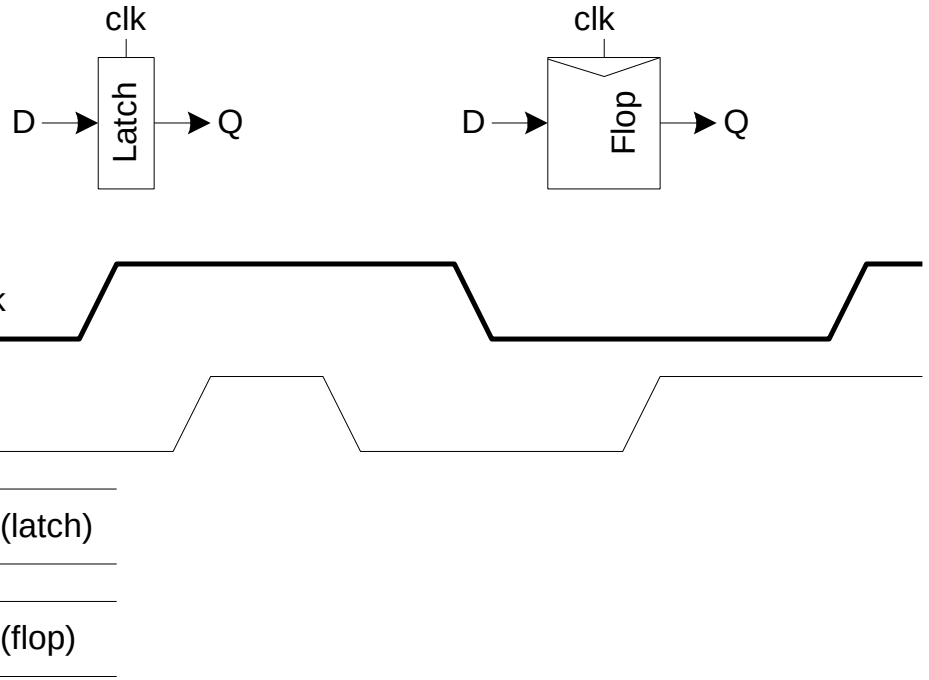
Pipeline

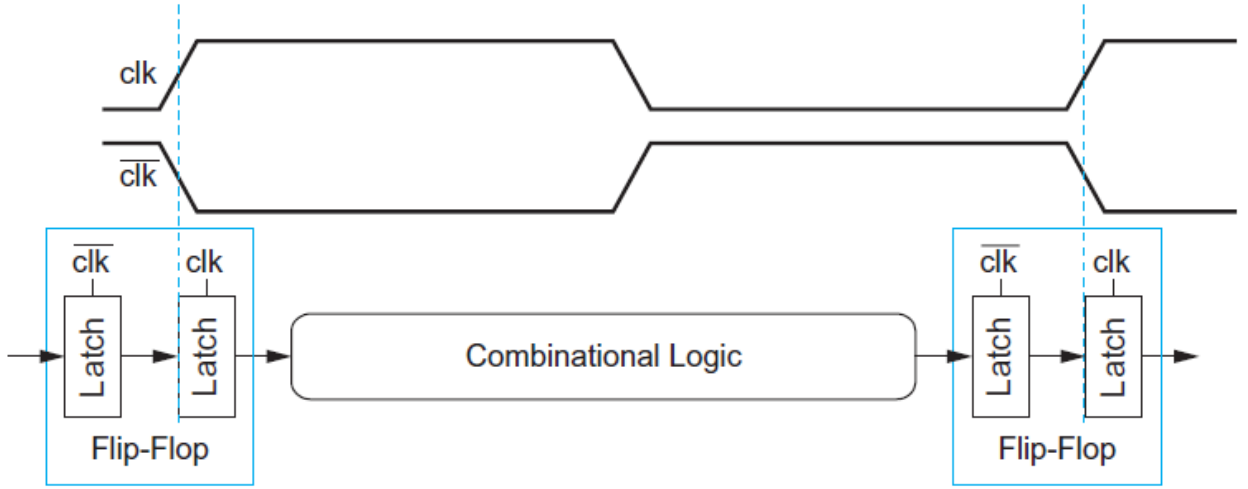
Static vs. Dynamic Logic

- Combinational logic circuits
  - Static circuits: no clock input
    - e.g., complementary CMOS, pseudo NMOS, and PTL
  - Dynamic circuits: requires clock input
    - e.g., domino logic
- Sequential logic circuits
  - Static storage: feedback used to retain output indefinitely
  - Dynamic storage: temporary charge on a capacitor
- Static/dynamic circuits can be sequenced using static/dynamic storage

Sequencing Elements

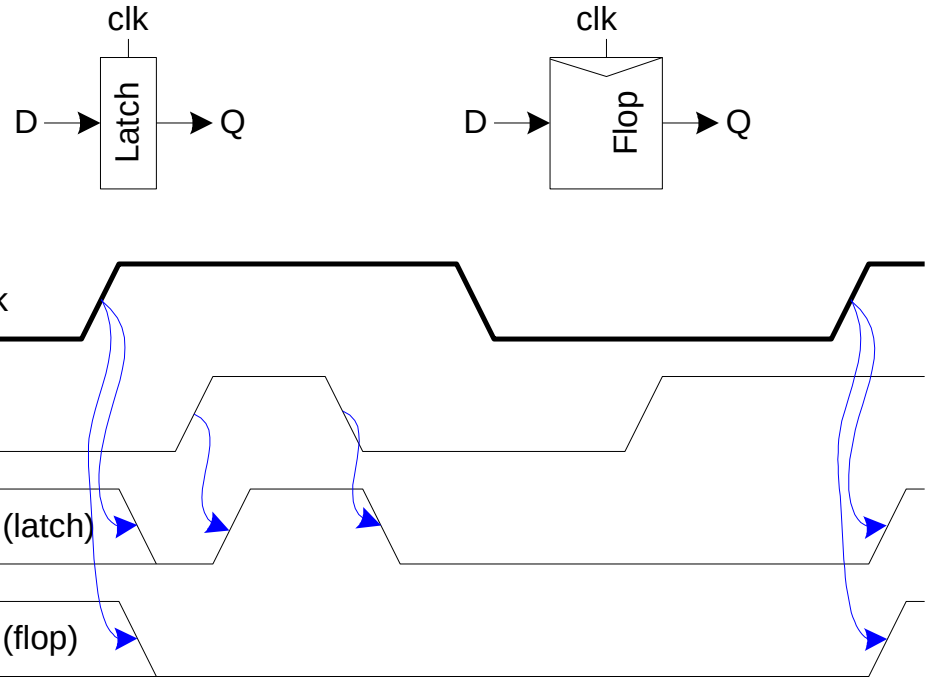
- **Latch**: Level sensitive
  - a.k.a. transparent latch, D latch
  - Transparent/opaque
- **Flip-flop**: edge triggered
  - a.k.a. master-slave flip-flop, D flip-flop, D register
  - Simply, a pair of latches using clk and its complement
  - Edge-triggered

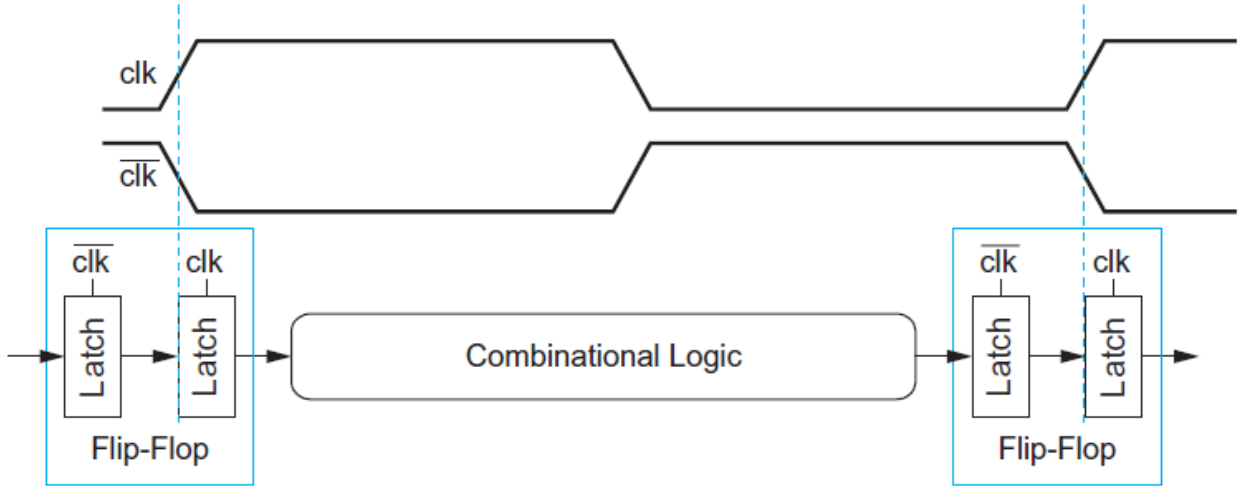




Sequencing Elements

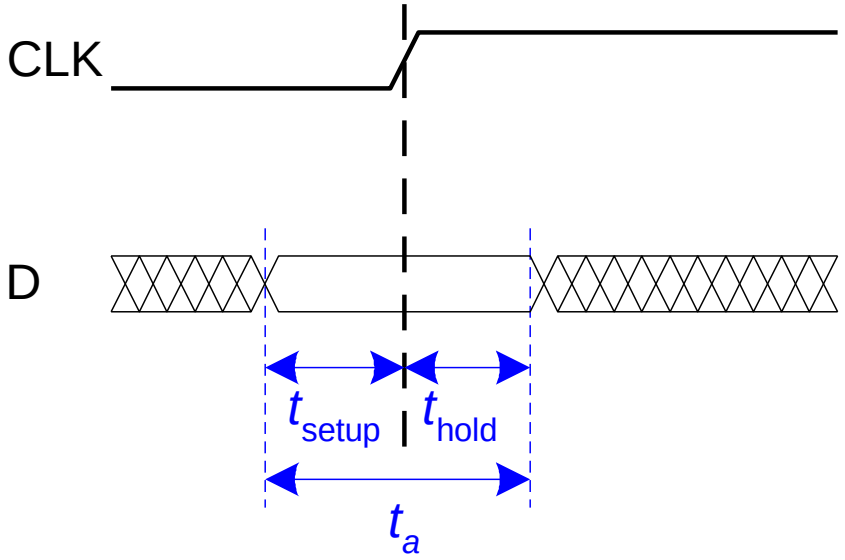
- **Latch**: Level sensitive
  - a.k.a. transparent latch, D latch
  - Transparent/opaque
- **Flip-flop**: edge triggered
  - a.k.a. master-slave flip-flop, D flip-flop, D register
  - Simply, a pair of latches using clk and its complement
  - Edge-triggered





Input Timing Constraints

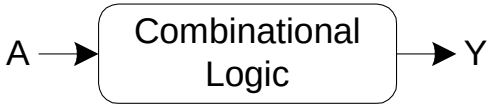
- Flip-flop samples *D* at clock edge
  - *D* must be stable when sampled
  - Similar to a photograph, *D* must be stable around clock edge
  - If not, metastability can occur
- **Setup time:** t_{setup} = time *before* clock edge data must be stable (i.e. not changing)
- **Hold time:** t_{hold} = time *after* clock edge data must be stable
- **Aperture time:** t_a = time *around* clock edge data must be stable ($t_a = t_{setup} + t_{hold}$)

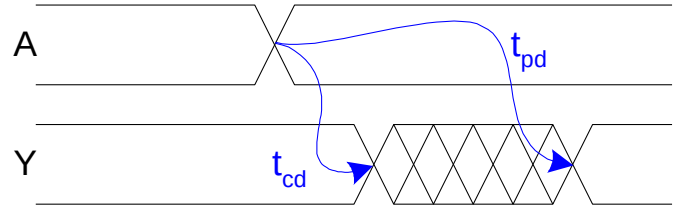


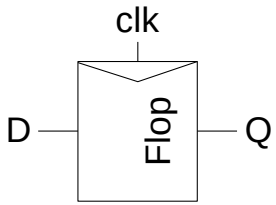
Timing Diagrams

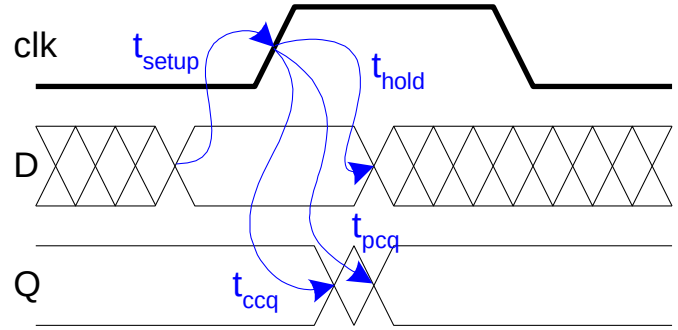
- Contamination delay: output may begin to change or glitch
- Propagation delay: output must have settled to final value

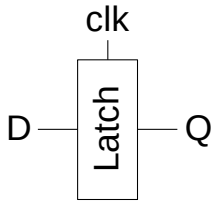
t_{pd}	Logic Prop. Delay
t_{cd}	Logic Cont. Delay
t_{pcq}	Latch/Flop Clk-to-Q Prop. Delay
t_{ccq}	Latch/Flop Clk-to-Q Cont. Delay
t_{pdq}	Latch D-to-Q Prop. Delay
t_{cdq}	Latch D-to-Q Cont. Delay
t_{setup}	Latch/Flop Setup Time
t_{hold}	Latch/Flop Hold Time
t_a	Aperture time

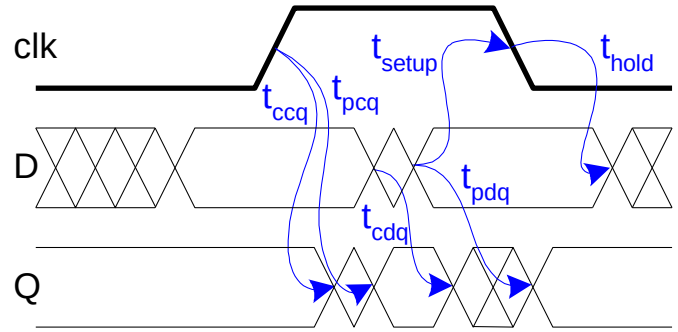






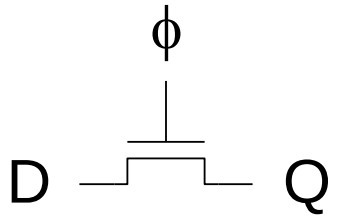






Latch Design

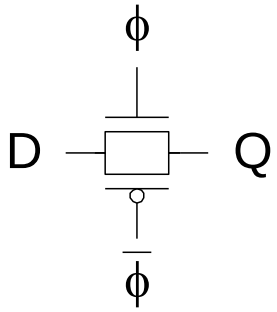
- Pass Transistor Latch
- Pros
  - + Tiny
  - + Low clock load
- Cons
  - V_t drop
  - dynamic (leakage)
  - nonrestoring
  - backdriving
  - output floating (noise sensitivity)
  - diffusion input



Used in 1970's

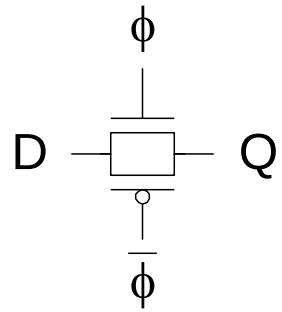
Latch Design

- Transmission gate
- Pros
  - + **No V_t drop**
  - + Tiny
  - + Low clock load
- Cons
  - **Requires inverted clock**
  - dynamic (leakage)
  - nonrestoring
  - backdriving
  - output floating (noise sensitivity)
  - diffusion input



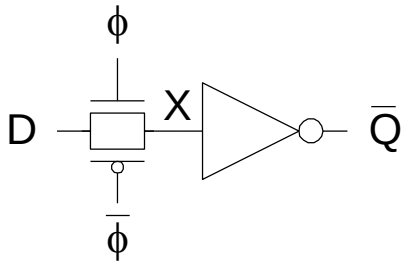
What is Bad About Diffusion Input?

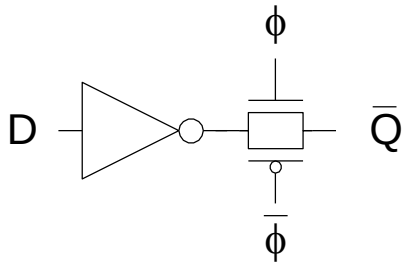
- Diffusion input is sensitive to noise
- Ex: power supply noise and/or coupling noise drove the input voltage below -Vt relative to GND seen by the transmission gate
  - Vgs now exceeds Vt for the NMOS
  - The transmission gate turns on
- If the latch stores 1, it could be incorrectly discharged to 0
- Also makes the delay harder to model with static timing analyzers
  - Delay will be dependent on the level of the input signal strong/weak
- However, exposing the diffusion input results in a faster latch
  - Can be used in datapaths where the inputs are carefully controlled and checked



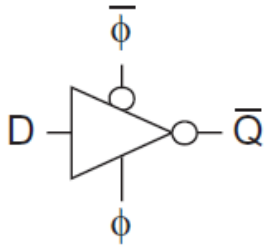
Latch Design

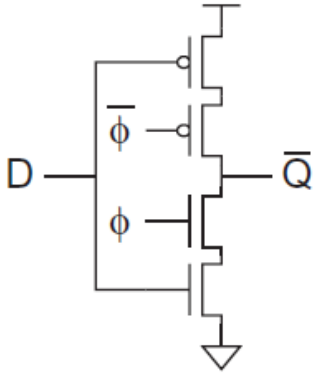
- Inverting buffer
  - + Restoring
  - + No backdriving
  - + Fixes either
    - Output noise sensitivity
    - Or diffusion input
  - Inverted output

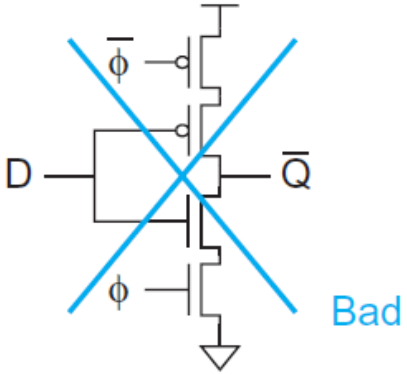




- Similar to a clocked tristate inverter: Clocked CMOS (C^2MOS)
  - But TG design is slightly faster

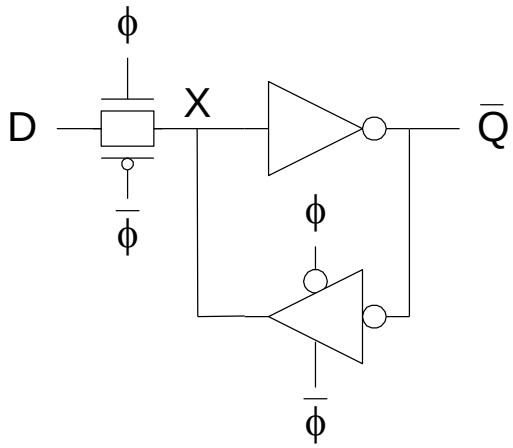






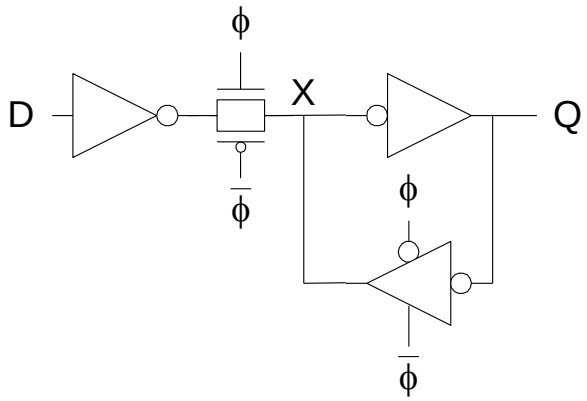
Latch Design

- Dynamic latches are sensitive to leakage
  - Requires min clock frequency (to refresh output)
  - Not compatible with clock gating
  - Leakage is worse in burn-in testing at elevated temperature and voltage
  - Not used in MPUs beyond $0.35\mu m$
- Tristate feedback
  - \+ Static
  - Backdriving risk (Output 'Q' may affect the state node 'X')
- Static latches are now essential because of leakage
  - But they are larger and slower than dynamic latches



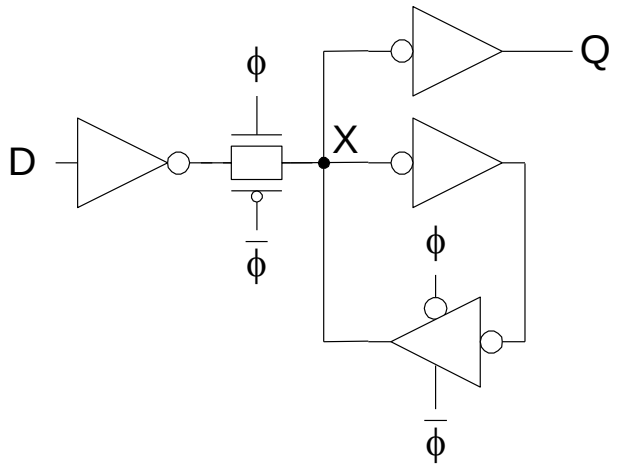
Latch Design

- Buffered input
  - + Fixes diffusion input
  - + Noninverting



Latch Design

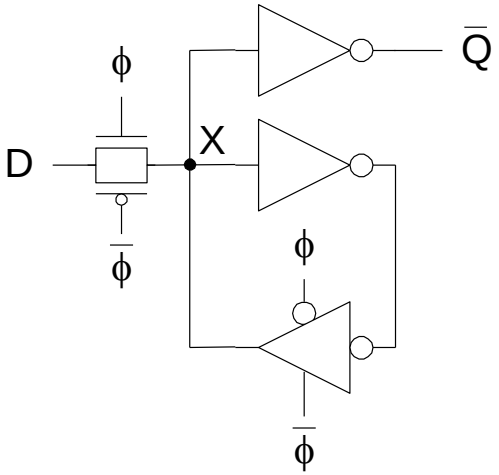
- Buffered output
  - \+ No backdriving
- **Widely used in standard cells**
  - \+ Very robust (most important)
  - \- Rather large
  - \- Rather slow (1.5 – 2 FO4 delays)
  - \- High clock loading
- Recommended for all but performance- or area-critical designs



Latch Design

- In semicustom datapath applications input noise can be better controlled

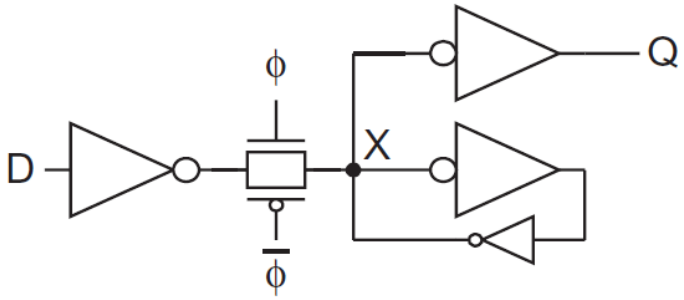
- Datapath latch
  - + smaller
  - + faster
  - ▪ unbuffered input

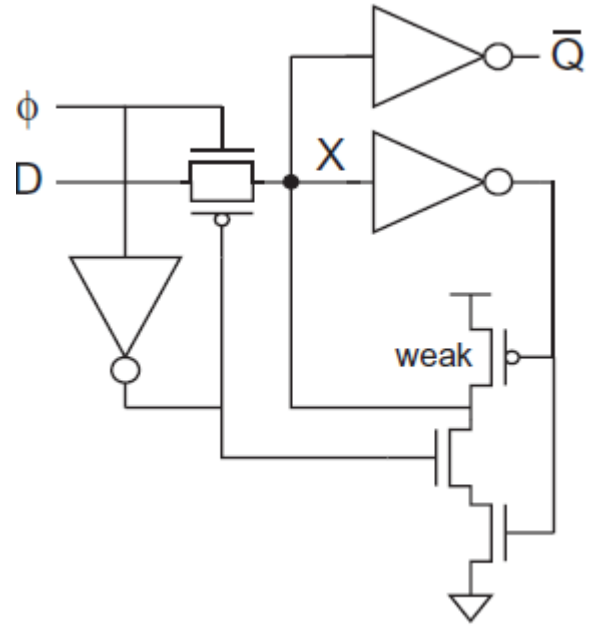


- Intel uses this as a standard datapath latch

Latch Design

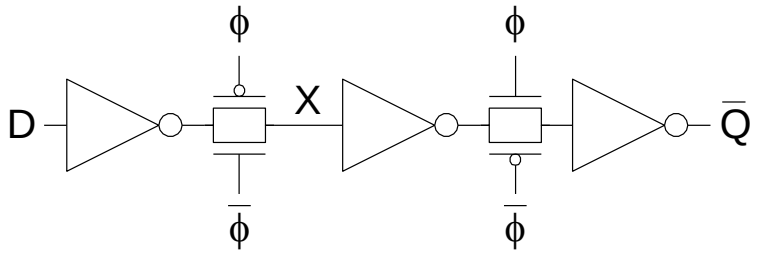
- Jamb latch
  - Uses a weak feedback inverter in place of the tristate
  - Saves two transistors
  - Reduces the clock load
  - But requires careful design to ensure that the tristate overpowers the feedback inverter in all corners
- Itanium 2 processor latch
  - Pulldown stack is clocked
  - Pullup is a weak PMOS transistor

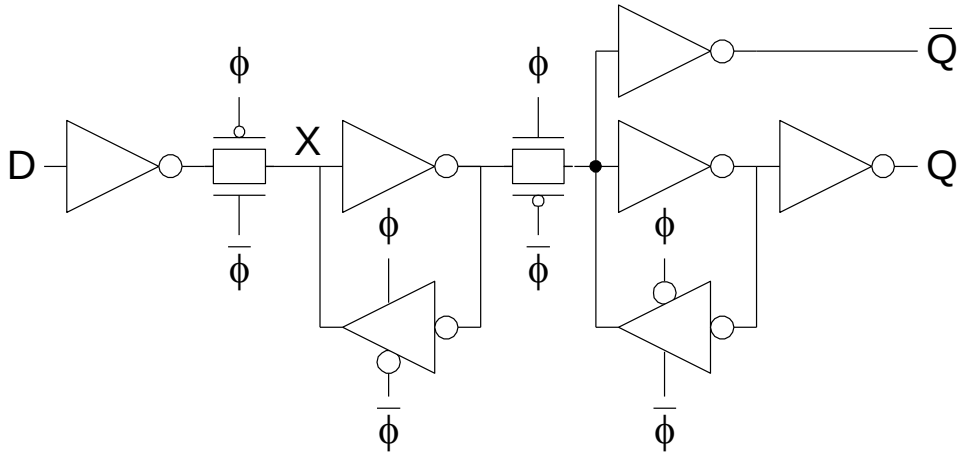




Flip-Flop Design

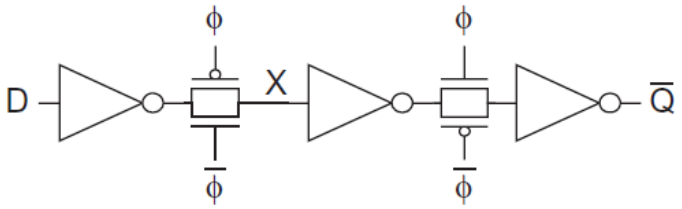
- Flip-flop is built as pair of back-to-back latches
- Dynamic
  - Leakage
  - Not used in MPUs beyond $0.35\mu m$
- Static
  - **Conventional flip-flop widely used in std cells**
  - Simple, robust, and energy-efficient
  - **But other alternatives may be faster**
  - In Artisan library, the clock is buffered and complemented in every flip-flop cell

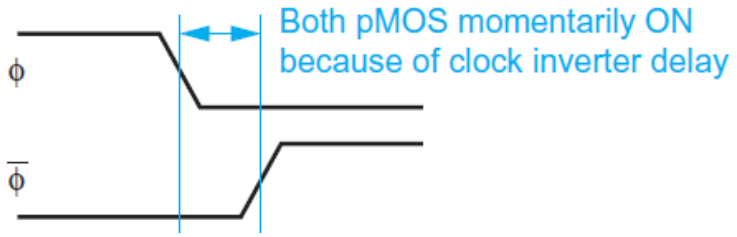




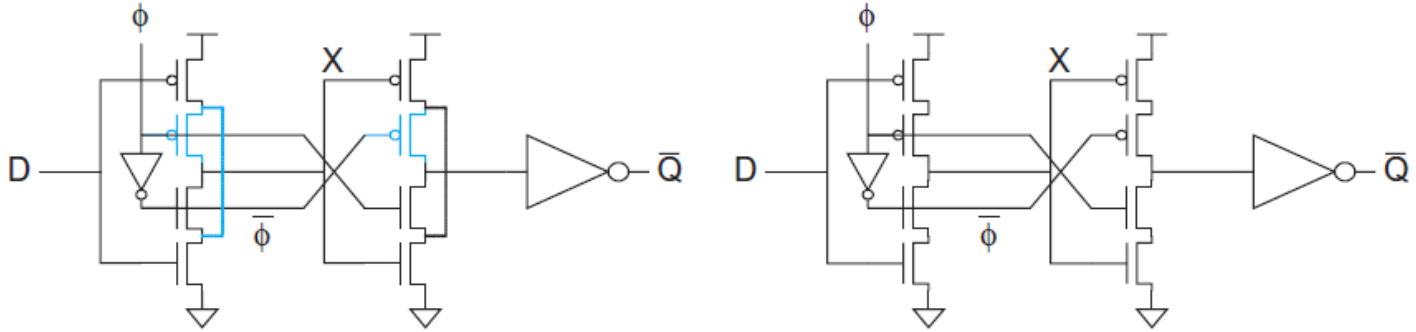
Flip-Flop Race Condition

- Skew between Φ and $\overline{\Phi}$: Both PMOS ON
- CMOS-TG: both latches are transparent → race condition





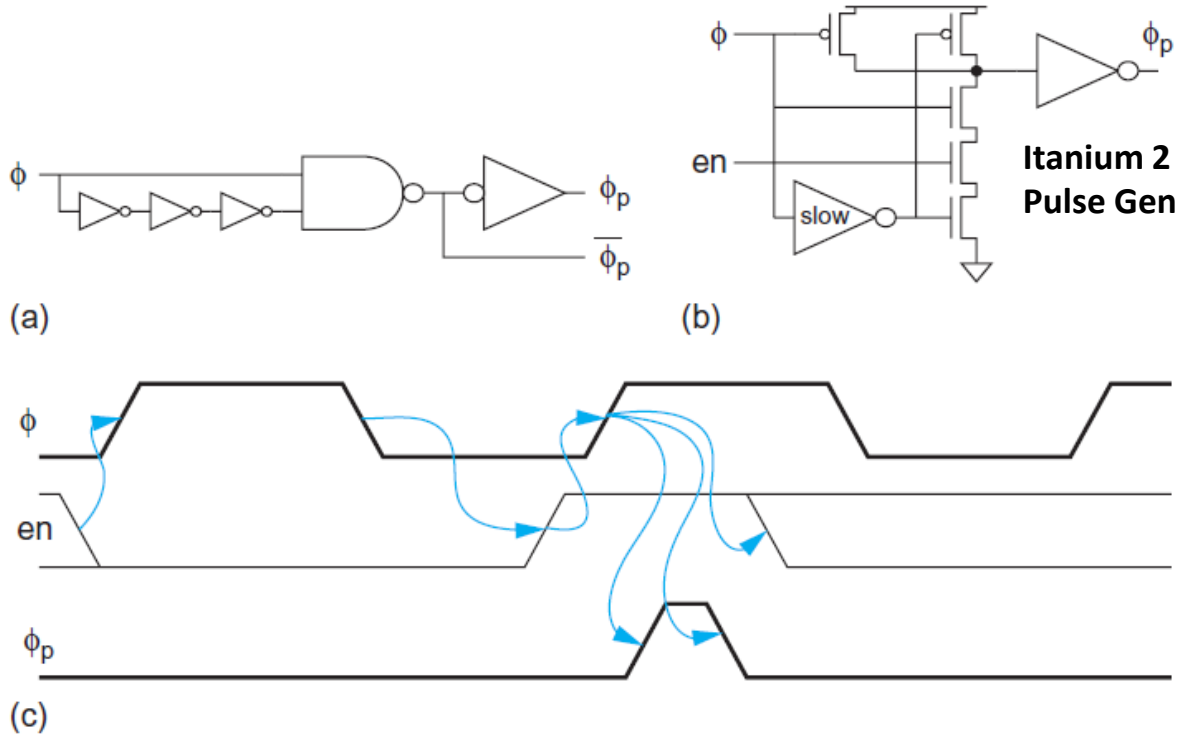
- C^2MOS: skew tolerant (you must go through NMOS and PMOS)
  - But clock must be reasonably sharp to avoid races
  - Even no. of logic stages can be placed between latches: NO RAce CMOS (NORA)



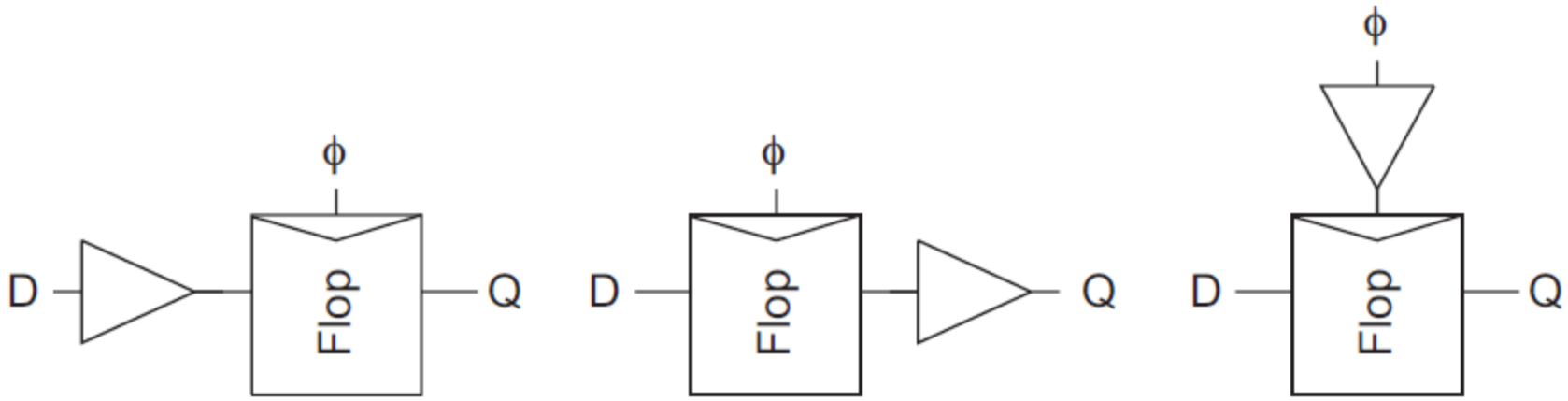
- Practically: CMOS-TG is slightly faster and is carefully designed to be safe

Pulsed Latch

- Conventional transparent latch driven by a clock pulse
- Pulse generator (one-shot) is required
  - Adds to the energy consumption
  - Shared across multiple latches for energy and area efficiency
- Faster than flip-flop, but increases hold time



Delay Trade-Offs

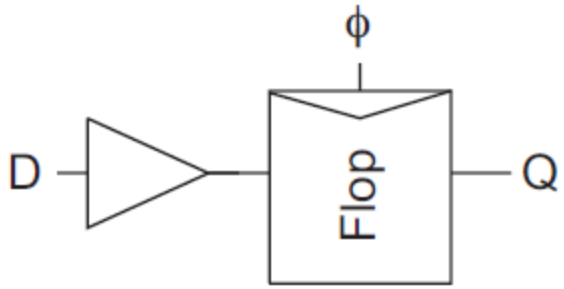


$$t'_{\text{setup}} = t_{\text{setup}} + t_{\text{buf}}$$

$$t'_{pcq} = t_{pcq}$$

$$t'_{\text{hold}} = t_{\text{hold}} - t_{\text{buf}}$$

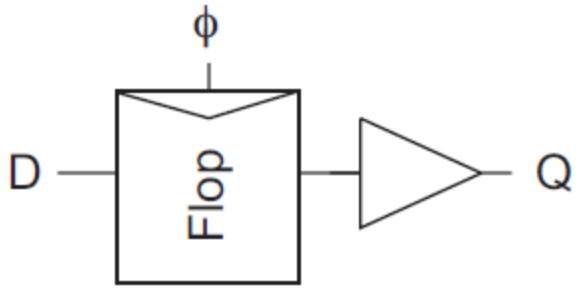
Delay Trade-Offs



$$t'_{setup} = t_{setup} + t_{buf}$$

$$t'_{pcq} = t_{pcq}$$

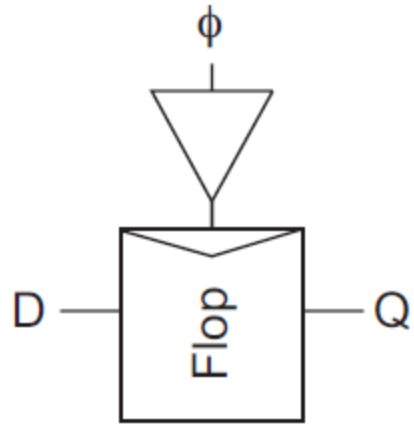
$$t'_{hold} = t_{hold} - t_{buf}$$



$$t'_{setup} = t_{setup}$$

$$t'_{pcq} = t_{pcq} + t_{buf}$$

$$t'_{hold} = t_{hold}$$



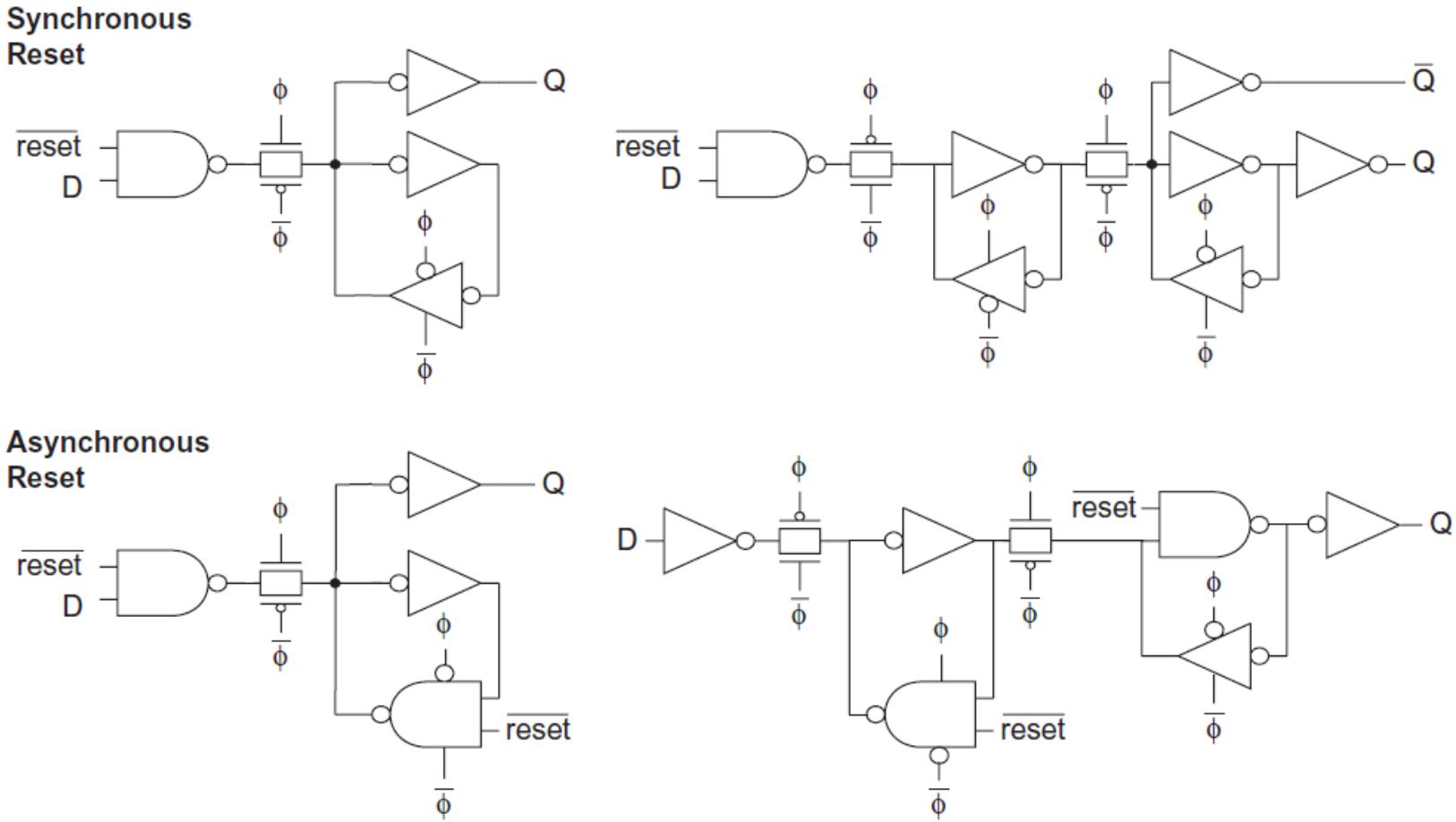
$$t'_{setup} = t_{setup} - t_{buf}$$

$$t'_{pcq} = t_{pcq} + t_{buf}$$

$$t'_{hold} = t_{hold} + t_{buf}$$

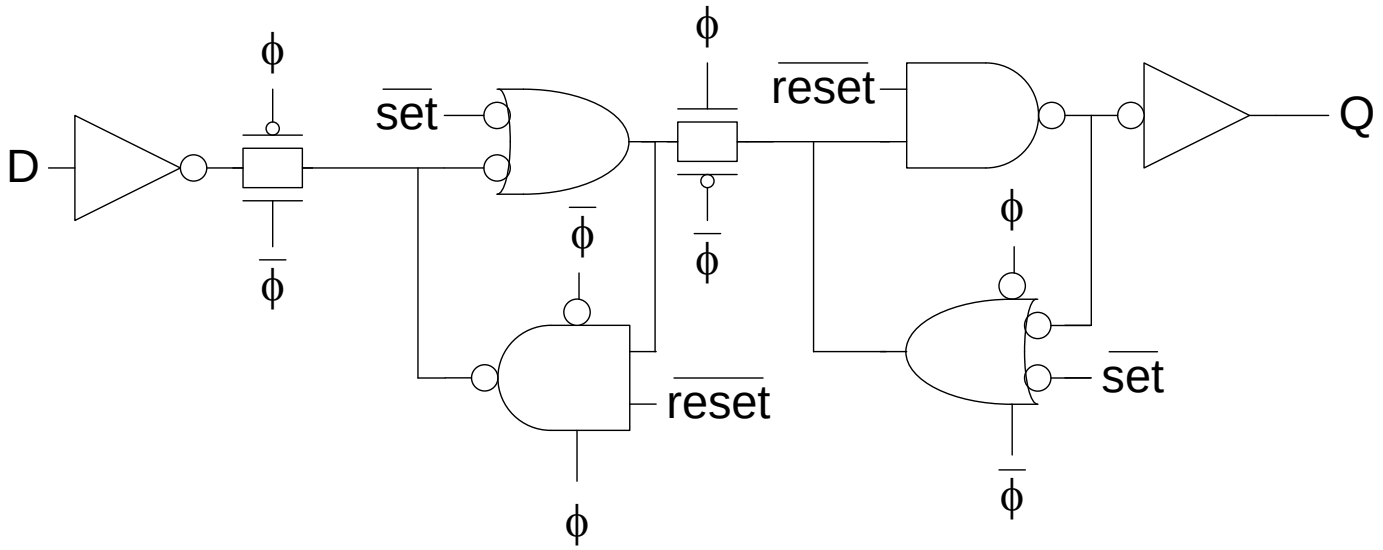
Reset

- Force output low when reset asserted
- Synchronous vs. asynchronous
- Tristate NAND gate = NAND gate + clocked TG



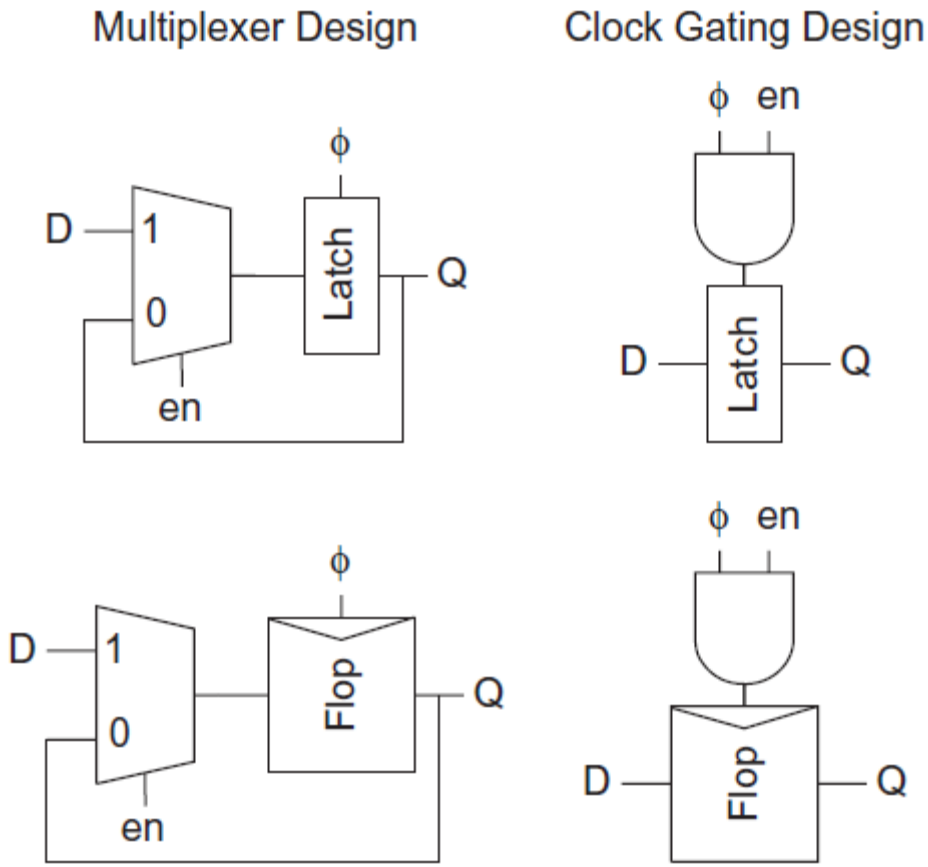
Set / Reset

- Set forces output high when enabled
- Flip-flop with asynchronous set and reset
- Recovery time: time between removing set/reset to clk input



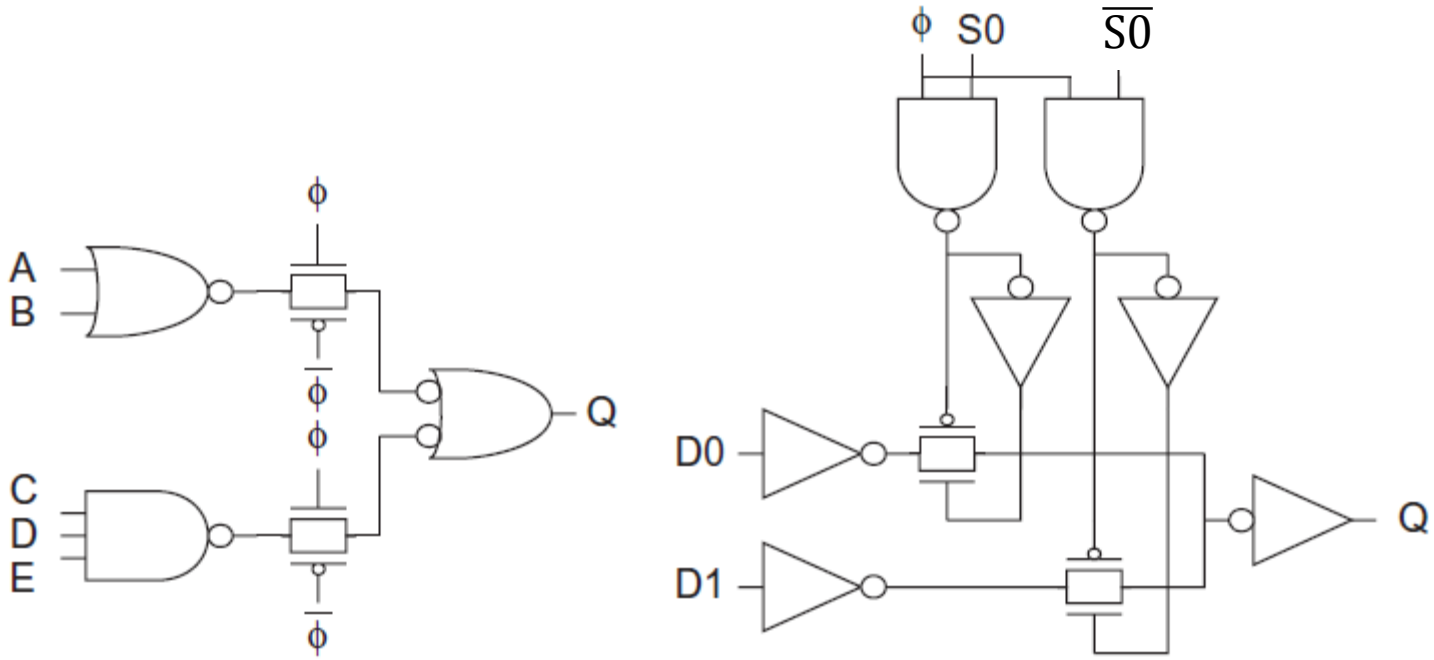
Enable

- Enable: ignore clock when en = 0
  - Mux: increases latch D-Q delay, area (safe choice)
  - Clock Gating: saves power, but hazardous, constrains en timing, adds skew



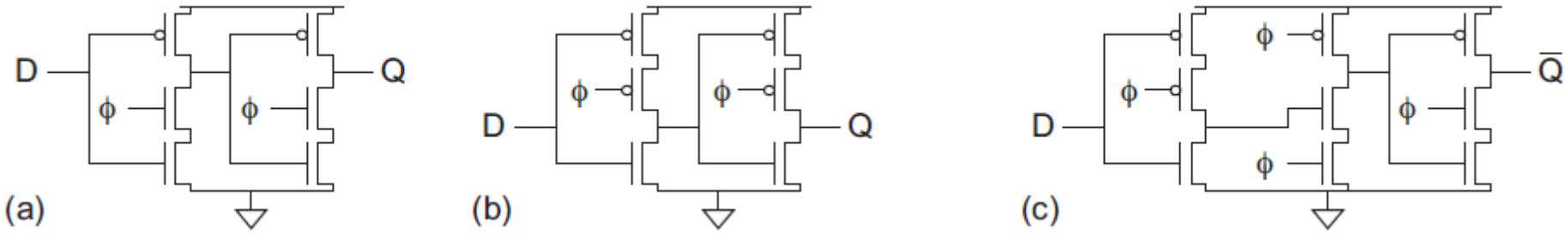
Logic into Latches

- Reduce sequencing overhead by incorporating logic into latches
- Replace inverters with gates that perform useful computation
- Mux-latch:
  - Integrates the multiplexer function with no extra delay from the D inputs to the Q
  - Setup time on the select inputs is relatively high.



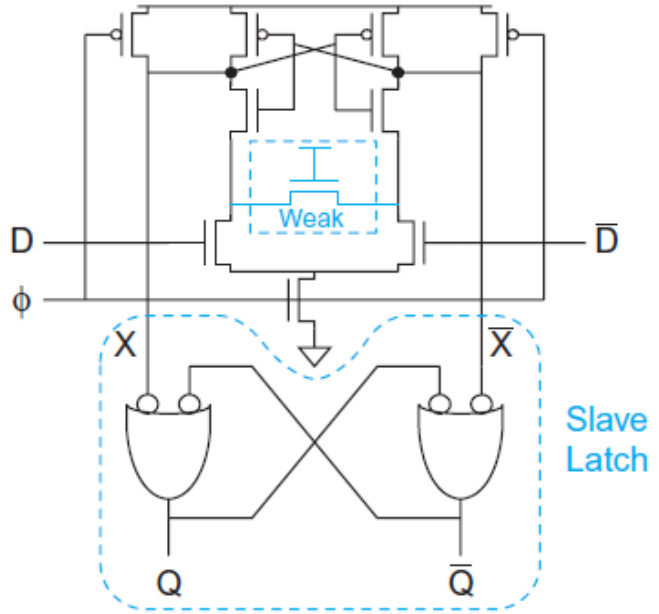
True Single-Phase Clock (TSPC) Latches

- CMOS-TG or C²MOS require Φ and $\overline{\Phi}$
- TSPC uses Φ only
- TSPC flip-flop produces a momentary low glitch on $\overline{Q}$ after the rising clock edge when D is low
  - Increases the activity factor of downstream circuits
- Not easy to staticize
- Clock must be reasonably sharp to avoid races
- TSPC is primarily of historical interest
  - But still useful for HS full custom digital, e.g., PLLs and serial links



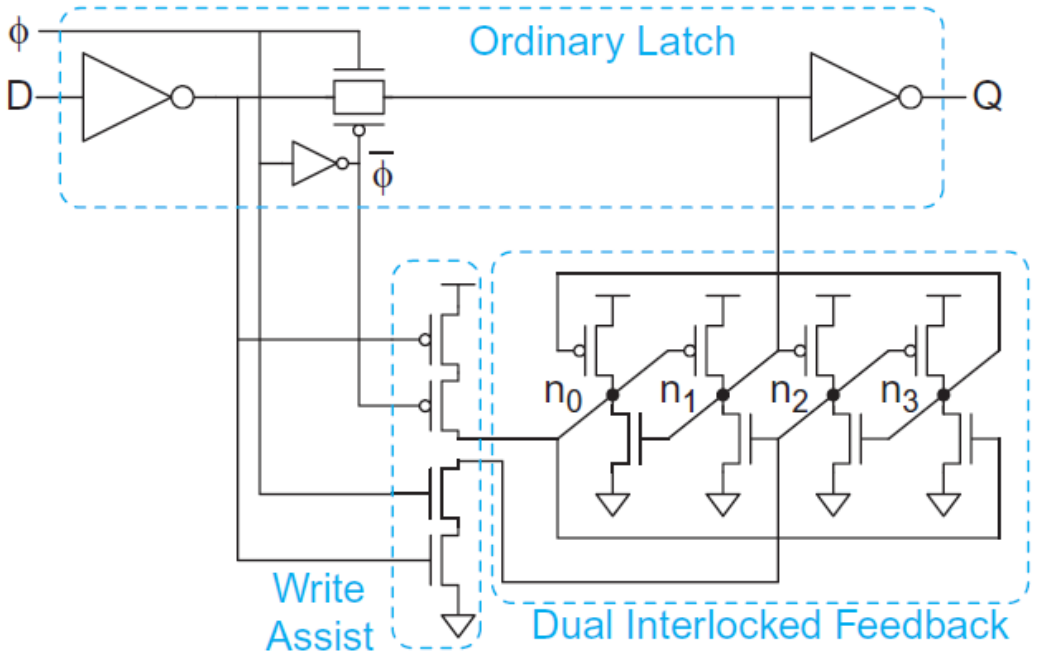
Differential Sense-Amplifier Flip-Flop

- Built from a clocked sense amplifier so that it can rapidly respond to small differential input voltage
  - Used in StrongARM MPU in the 1990s (a.k.a. StrongARM latch)
- CLK = 0: internal nodes X and $\bar{X}$ precharge.
- CLK = 1: one of X and $\bar{X}$ is pulled down
- SR latch: slave stage, captures output and holds it during Precharge (static output)



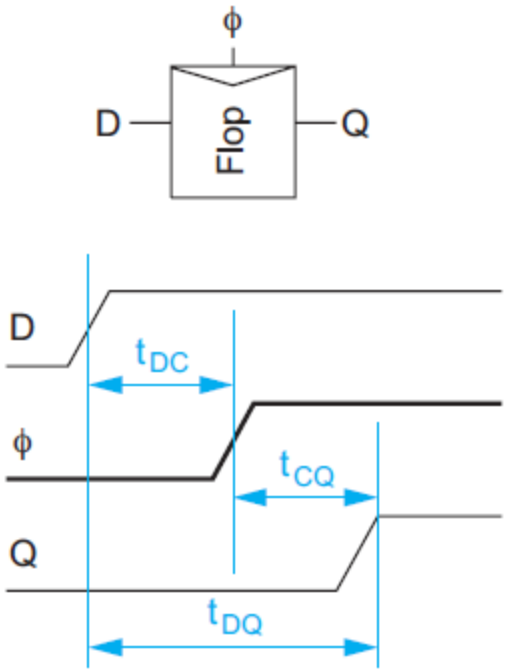
Radiation-Hardened Flip-Flops

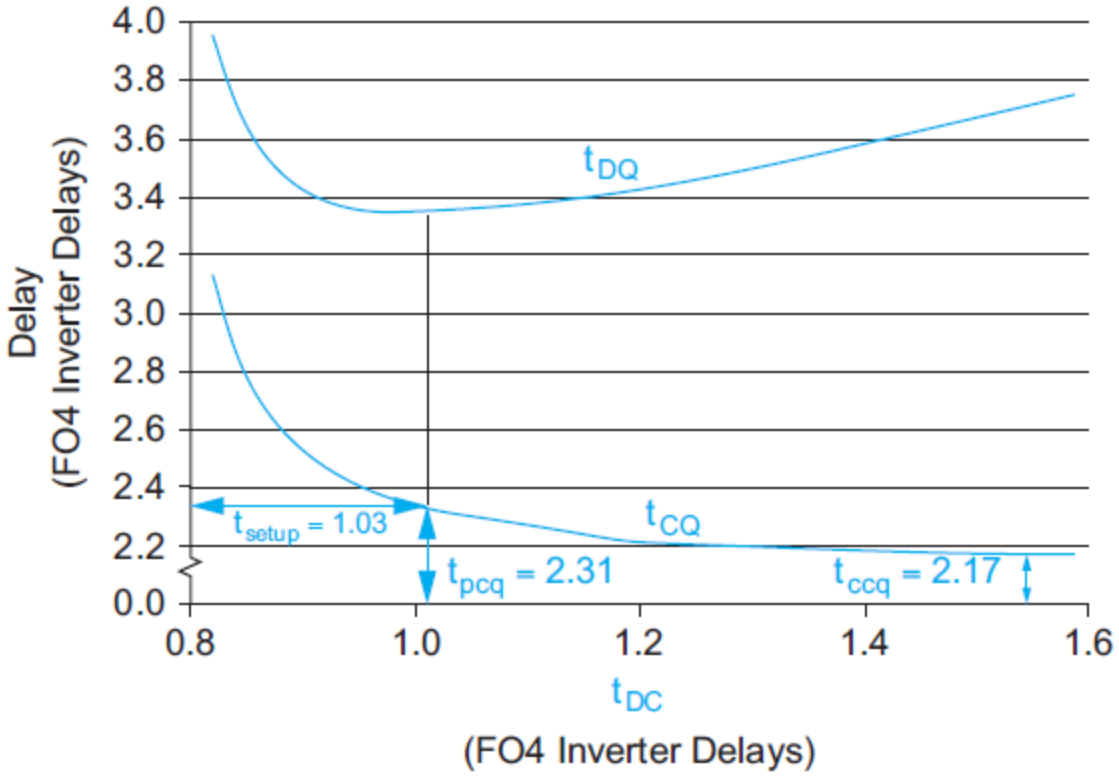
- Soft errors are random nonrecurring errors triggered by radiation striking a chip (mostly from cosmic radiation)
  - Critically important for space applications
- Traditionally affected DRAMs, but becoming more important for registers as charge diminishes (scaling dimensions and voltage)
- If one node (n1 to n4) is disturbed, the feedback will restore it



Characterizing Sequential Circuits

- Read Section 10.4.2 in the textbook: Characterizing Sequencing Element Delays
- t_{DQ} is min when the slope of t_{CQ} is -1 (increase/decrease in x is counteracted by an equal decrease/increase in y)

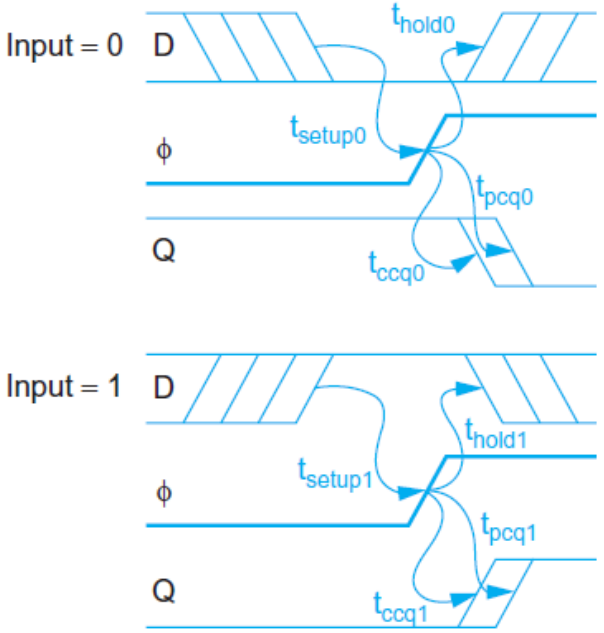


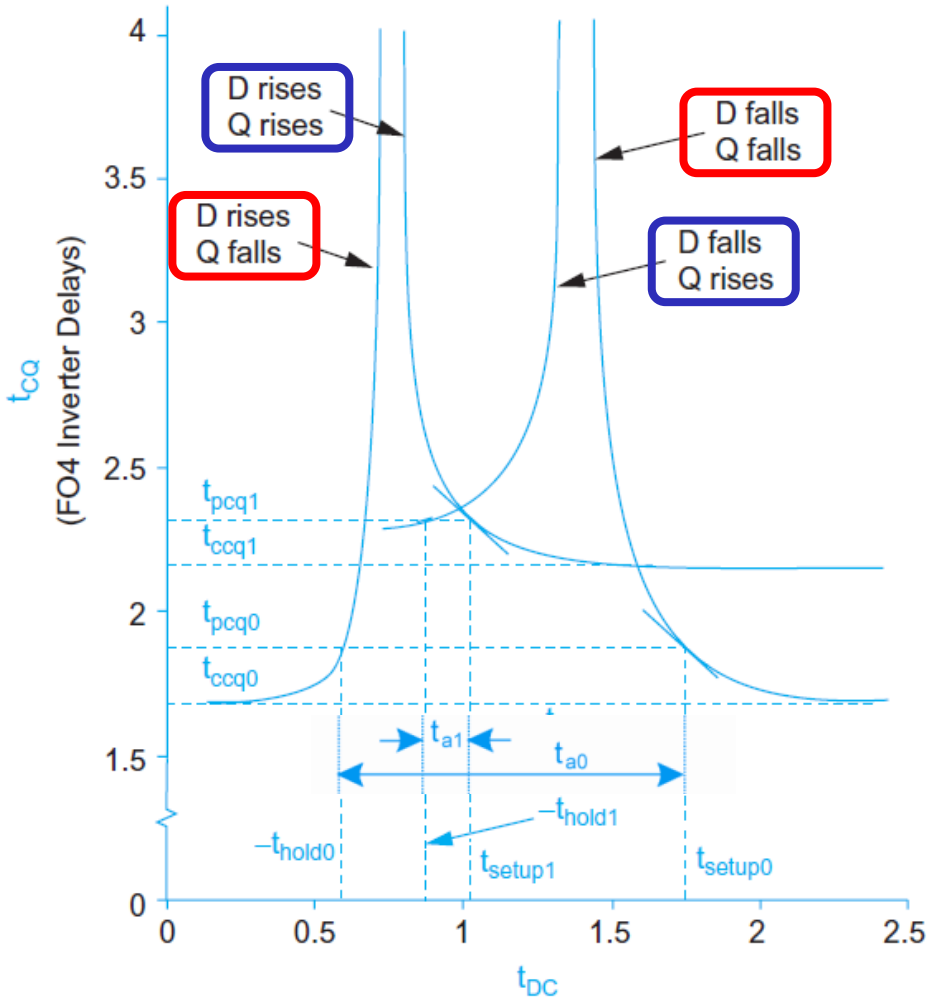


Characterizing Sequential Circuits

- Aperture time: The time that input must remain '0' or '1' to be correctly sampled

$$t_{a0} = t_{setup0} + t_{hold0}$$
$$t_{a1} = t_{setup1} + t_{hold1}$$





Thank you!

Characterizing Sequential Circuits

- Latch delay vs. data arrival time