

東北大学 工学部 機械知能・航空工学科
2017年度 5セメスター・クラスC3 D1 D2 D3

計算機工学

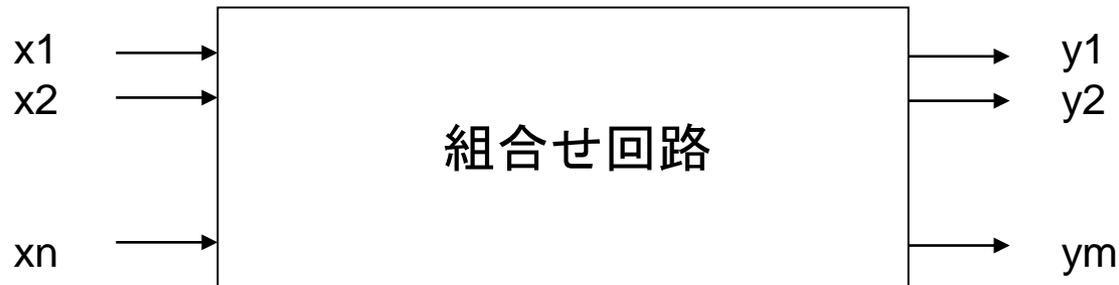
11. 順序回路の基礎 (教科書4章)

大学院情報科学研究科

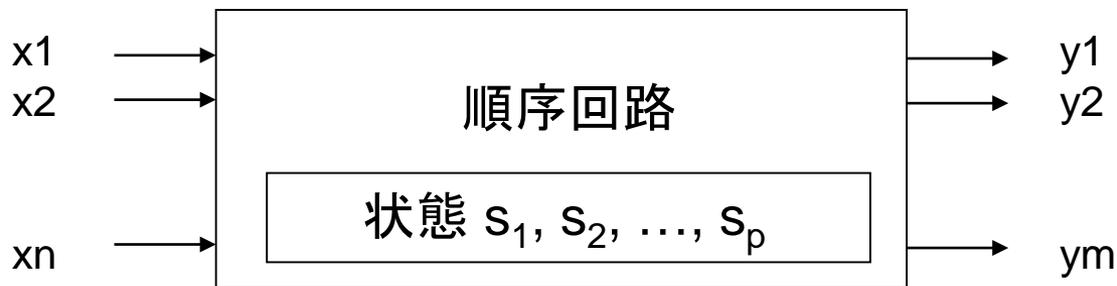
鏡 慎吾

<http://www.ic.is.tohoku.ac.jp/~swk/lecture/>

組合せ回路と順序回路



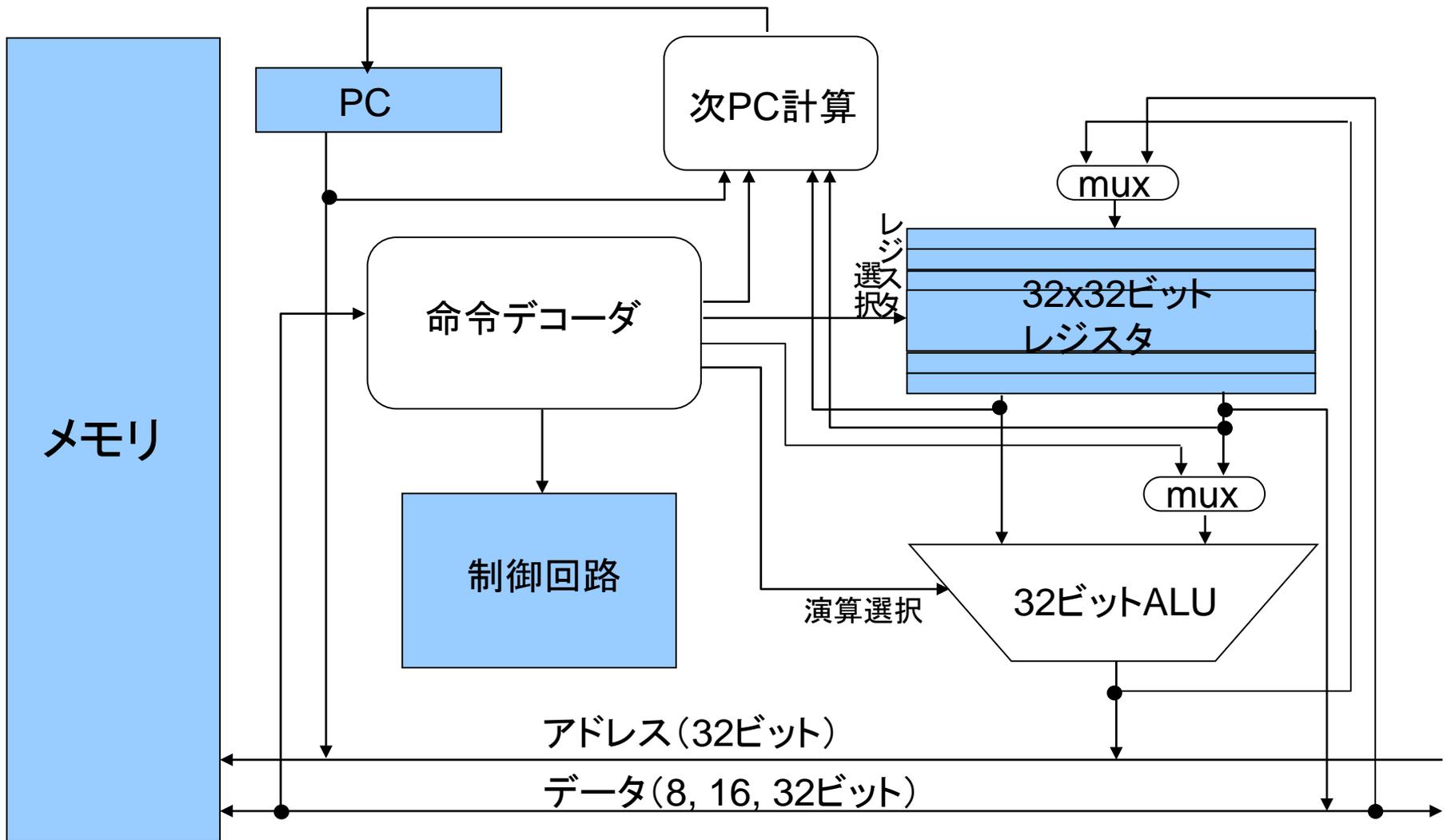
組合せ回路: 出力は, その時点の入力の組合せのみで決まる



順序回路: 出力は, 現在までにどんな入力がどんな順序で与えられたかによって決まる

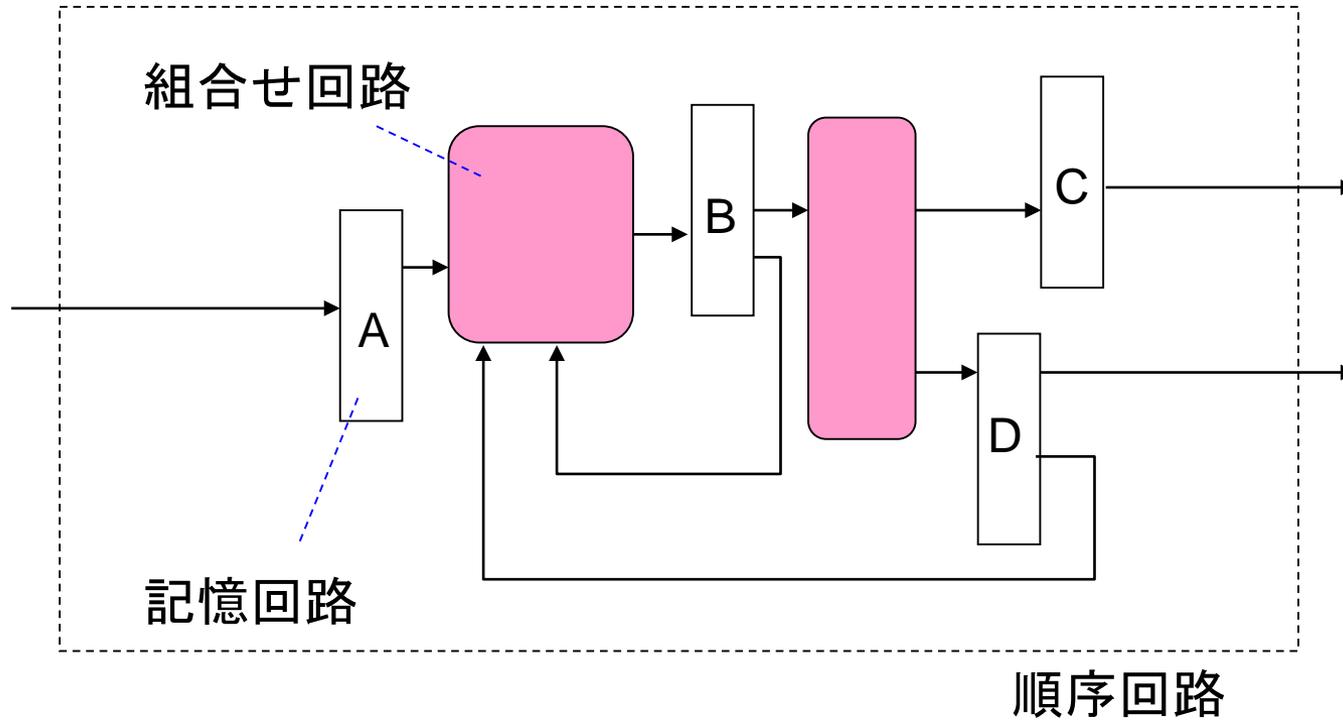
回路内に**内部状態**(記憶)が必要

復習: MIPSの構造



- 記憶素子を含む回路はすべて順序回路
- 全体も大きな順序回路と言える

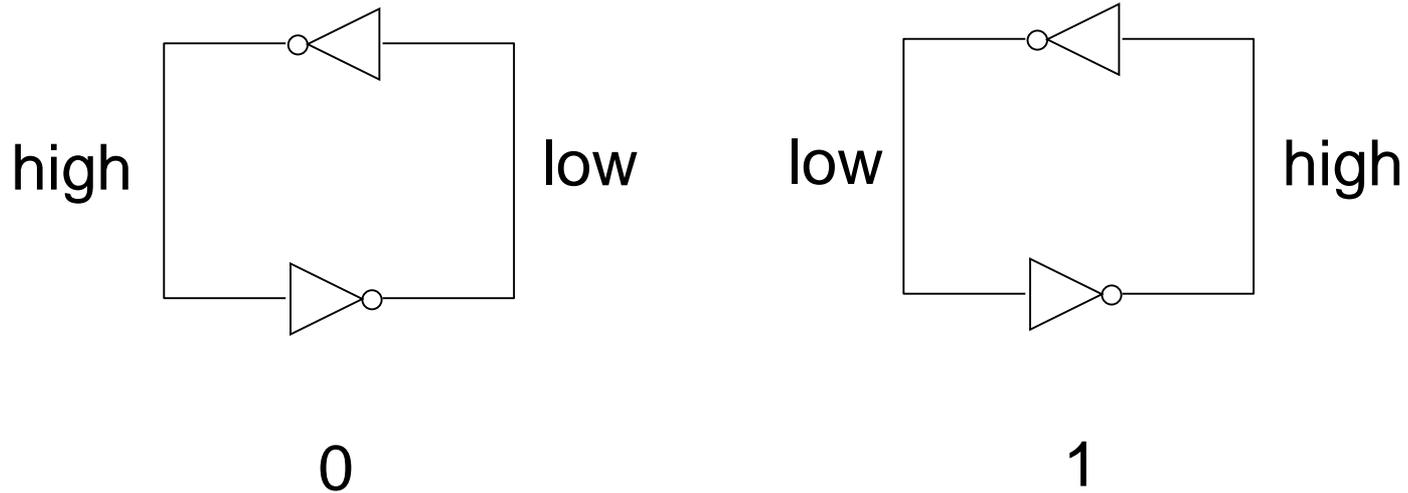
(非同期式)順序回路



- 順序回路 = 記憶回路と組合せ回路のネットワーク

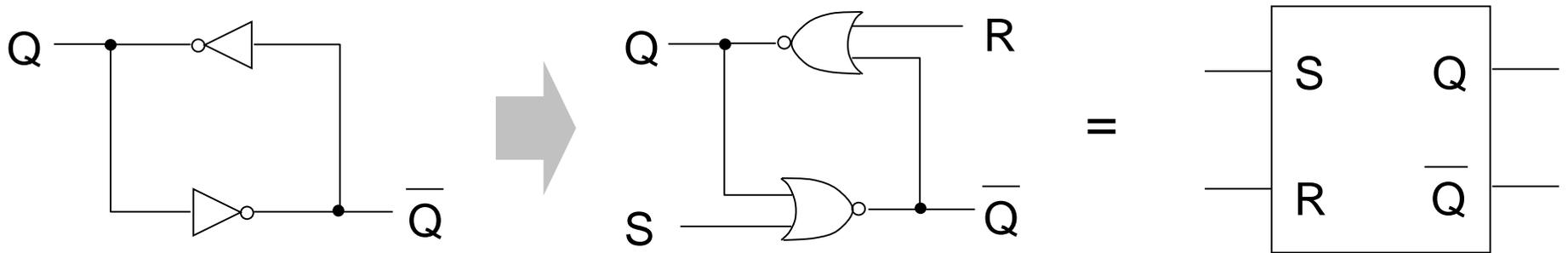
記憶の原理

2つの安定な物理状態を持つ系は、
1ビットの情報を記憶することができる



問題は、どのように状態を変化させるか

SR (Set-Reset) フリップフロップ (FF)



- NOT ゲートの代わりに NOR ゲートを使うと, 入力 S, R を 1 にすることで信号のループを断ち切って Q を変えることができる

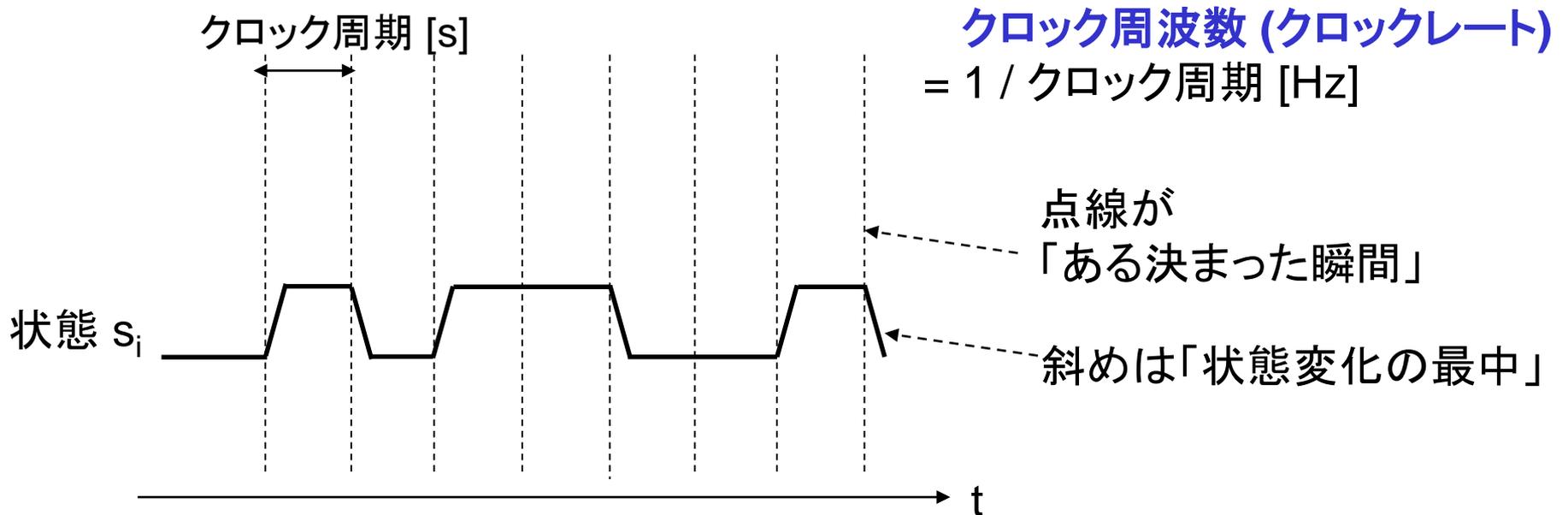
S	R	Q
0	0	保持
0	1	0
1	0	1
1	1	不定

- 問題点:
- $S = R = 1$ のときの動作が未定義
 - 非同期動作

同期式順序回路

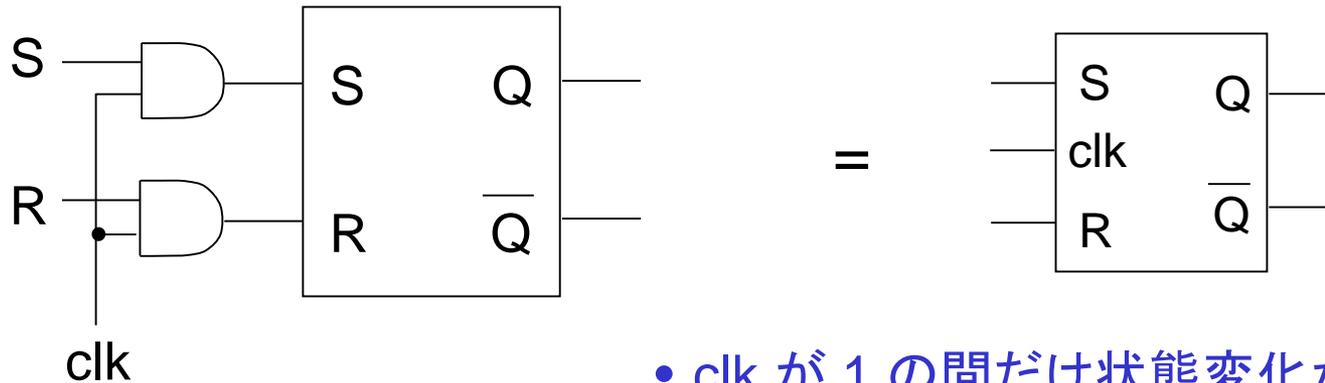
- 状態変化がいつでも起き得る回路は、設計・動作保証が困難
- ある決まった瞬間にしか状態変化が起きないように制限する
- 「ある決まった瞬間」はクロック信号により周期的に与える
(連続時間システムから離散時間システムへ)

タイミングチャート (横軸が時間, 縦軸が信号レベル)



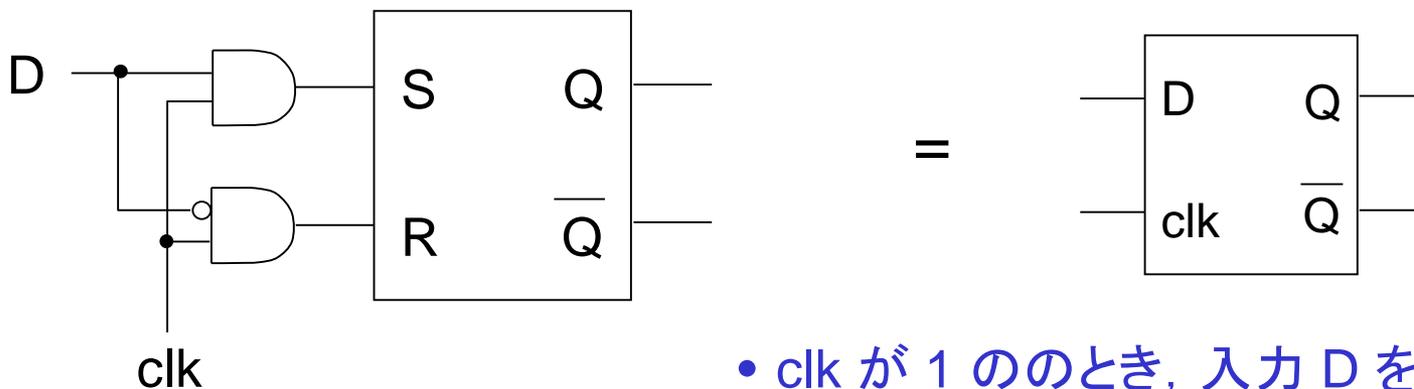
クロック入力付きFF

SRフリップフロップ (SRラッチ)



- clk が 1 の間だけ状態変化が許される
- 入力信号が3本もあるのは冗長

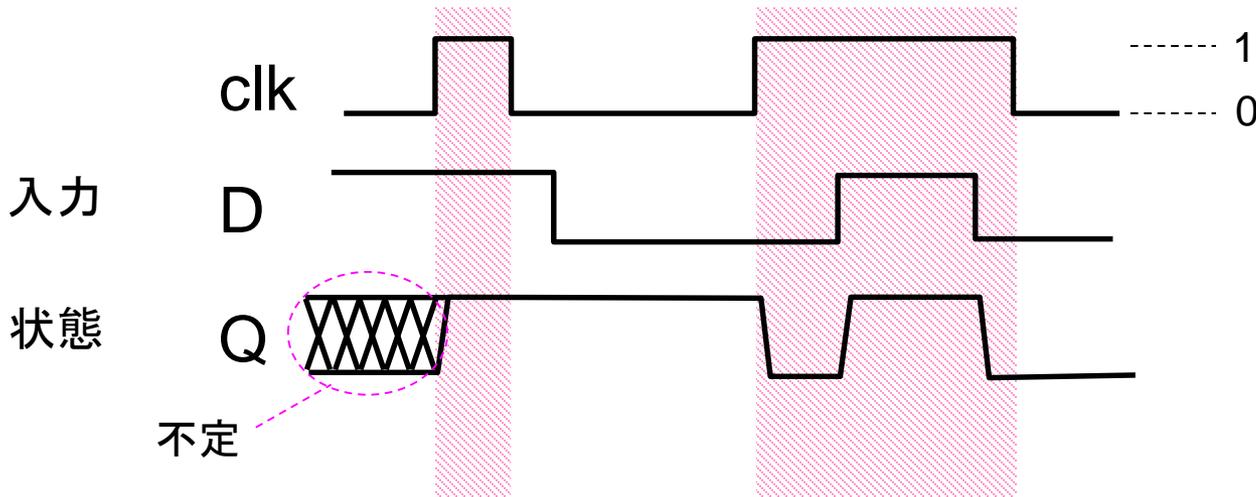
Dフリップフロップ (Dラッチ)



- clk が 1 ののとき, 入力 D を記憶する
- D は Delay または Data の意とされる

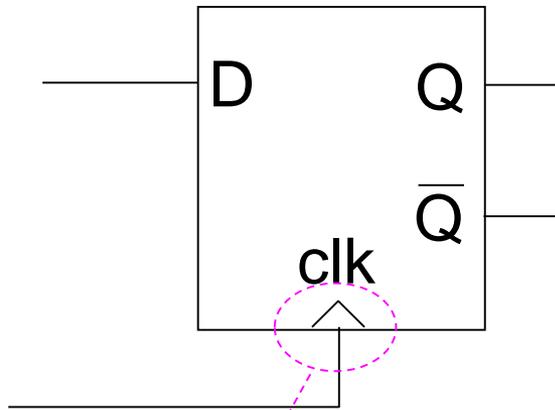
D-FF (Dラッチ) の動作

例えば clk が(あまり「クロック」っぽくないが)以下のように動いたとすると



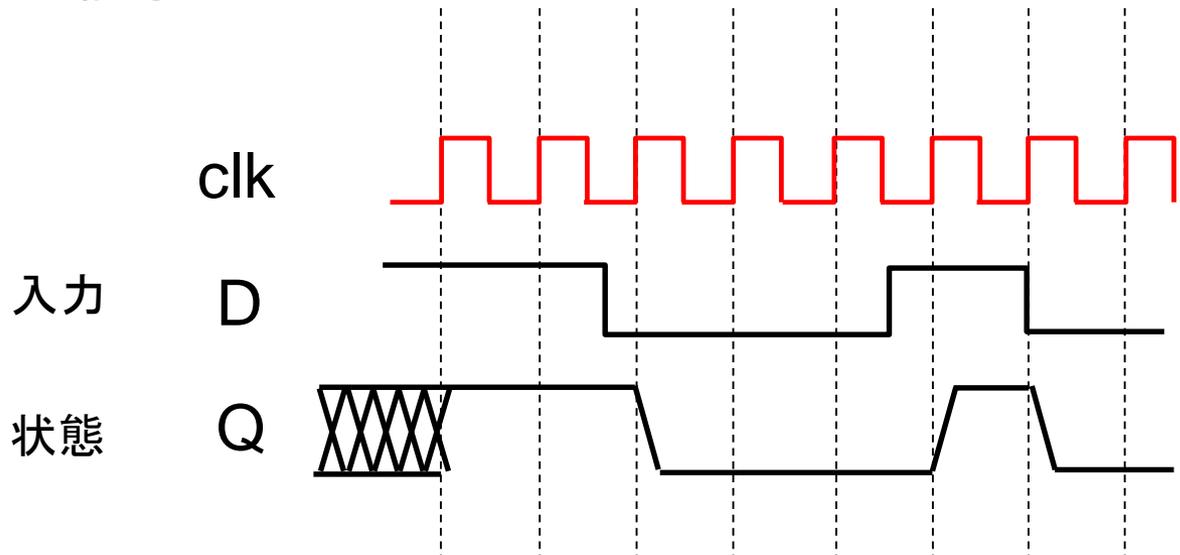
- 問題点:
- クロック信号が1の間は, 入力は出力に筒抜け
 - 1 から 0 に戻るときに確定
 - 同期式回路に使おうとすると, 「クロック信号が1の間」の長さは,
 - 長すぎてもダメ: 状態変化の結果が入力に影響してしまう
 - 短すぎてもダメ: FF自体の動作が間に合わない

エッジトリガ FF

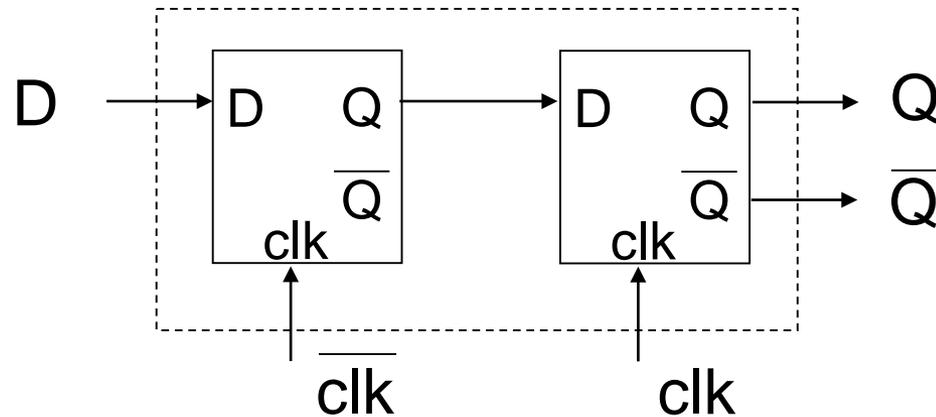


山型はエッジトリガ入力の記号

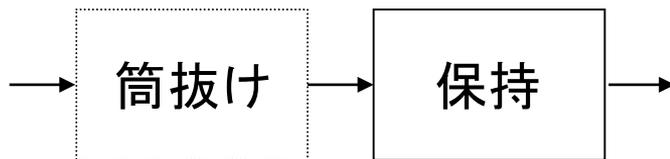
- クロック信号の**立上りの瞬間(直前)**の値を記憶する
- その瞬間以外に状態変化が起きることはなく、信号が筒抜けになることもない
- エッジトリガ型と呼ぶ
 - 対して、クロックが 1 か 0 かに応じて動作が変わるものをレベルセンシティブ型と呼ぶ



エッジトリガ D-FFの構成例 (マスタースレーブ型)

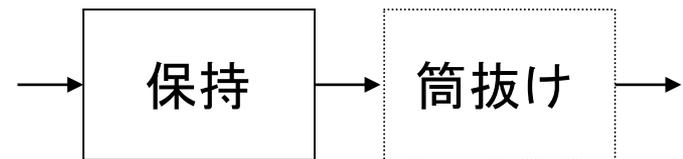


clk = 0 のとき:



後段のラッチは前の値を保持。
前段のラッチは開き、次に保持すること
になる情報を後段に素通しにしている

clk = 1 のとき:



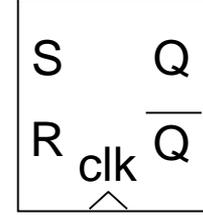
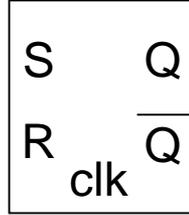
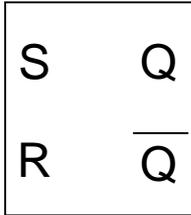
クロックが立ち上がると、前段のラッ
チが閉じて**その瞬間の値を記憶**し、
後段のラッチは開いて素通しになる

非同期

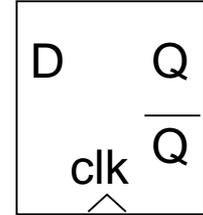
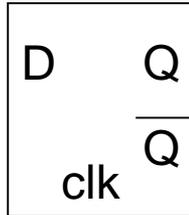
レベルセンシティブクロック入力

エッジトリガクロック入力

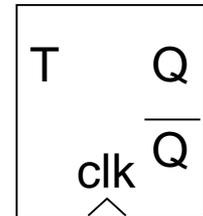
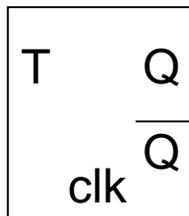
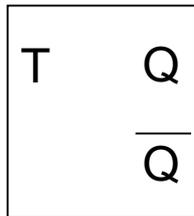
SR



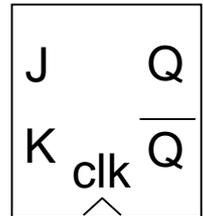
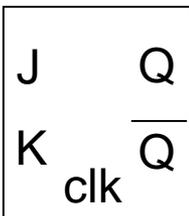
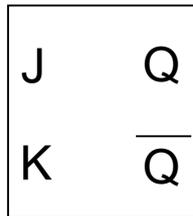
D



T



JK



いろいろなフリップフロップと関連用語

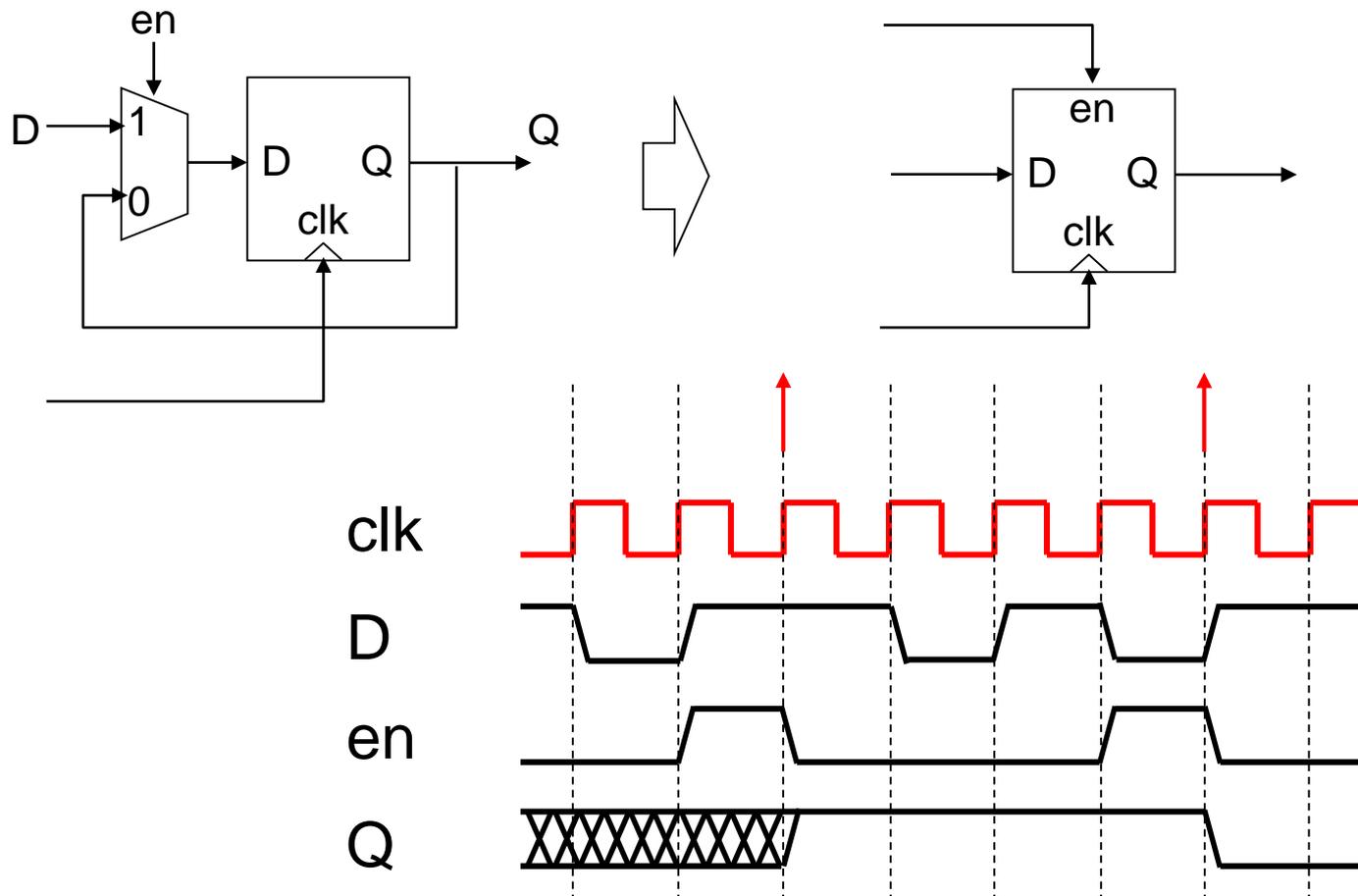
- 前ページのものをすべてフリップフロップと呼ぶ流儀と、エッジトリガのものだけフリップフロップと呼び、非同期あるいはレベルセンシティブなものはラッチと呼ぶ流儀がある
- 特定の構成方法のもののみをエッジトリガ型と呼び、マスタースレーブ型とは区別する流儀もある
- レベルセンシティブクロック入力をクロックとは呼ばず、イネーブル信号と呼ぶ流儀もある

- T-FF は、入力 T が 1 になると Q が反転する (Toggle)
- JK-FF は、両入力が 1 のときに Q が反転するように SR-FF を改良したもの (J と K の由来は不明)

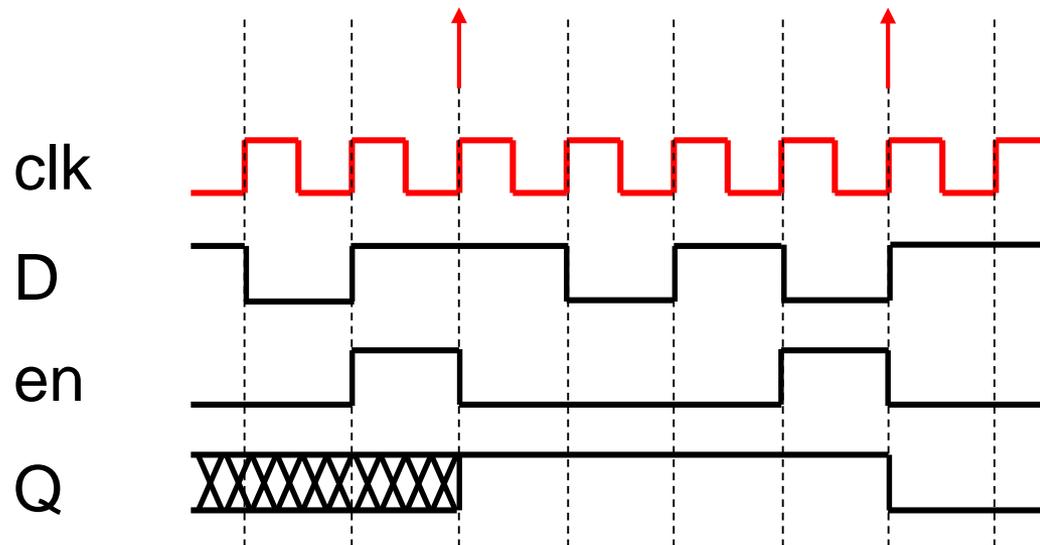
コンピュータの理解には、エッジトリガ型 D-FF だけでほぼOK。以降、D-FF といったらこれを指す

イネーブルつきD-FF

- 毎クロック必ず記憶するのが便利とは限らない
- $en = 1$ であるようなクロック立上り時にだけ入力を記憶する
(あるクロック立上りで記憶するかどうかはその瞬間の en で決まる)

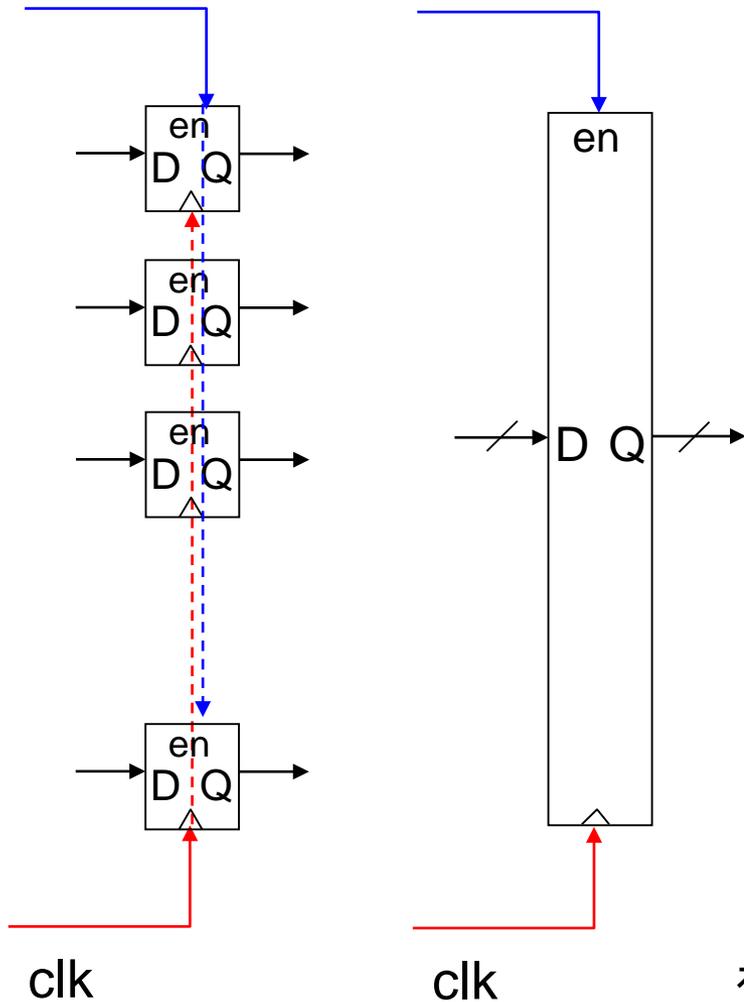


立上り・立下り時間を無視して描かれたチャートの解釈

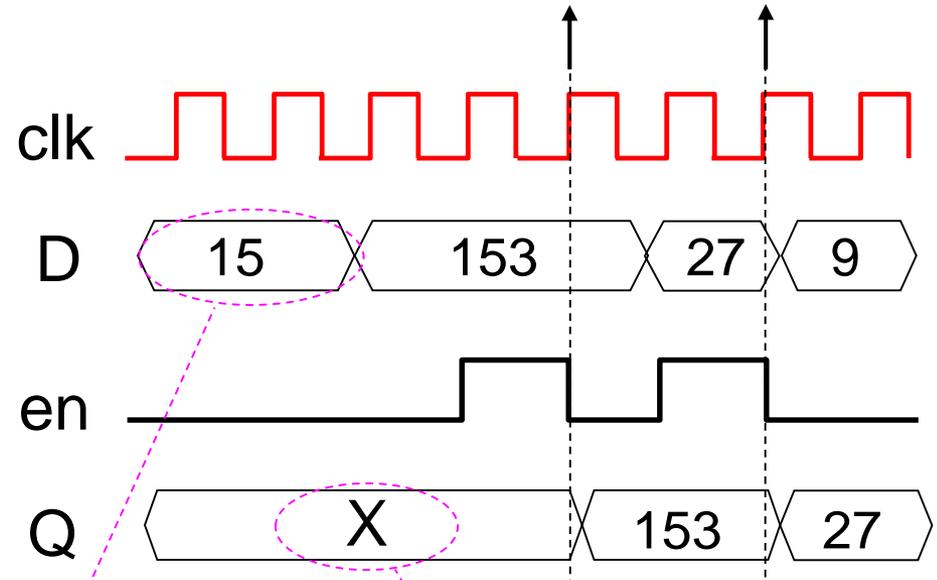


- 上図のように描かれた場合「クロック立上りの瞬間の D や en の値」はどこを見ればよいのか?
→ クロック立上りの直前を見る
- D も en も同じ clk に同期した回路から生成されていると考える(完全同期式回路). したがって D や en の変化は clk の立上りに先立って起きることはない

レジスタ



D-FFを n 個並べる \rightarrow n ビットレジスタ

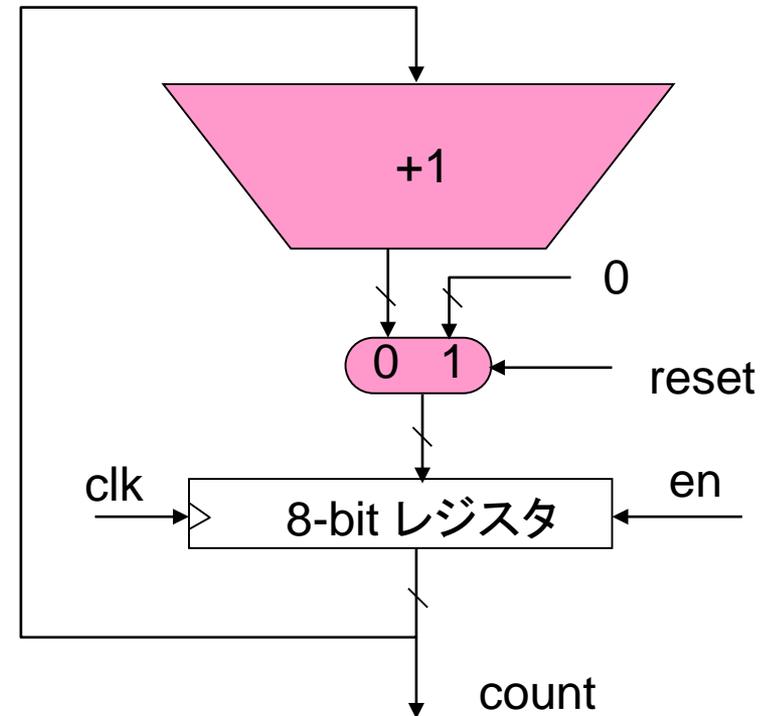
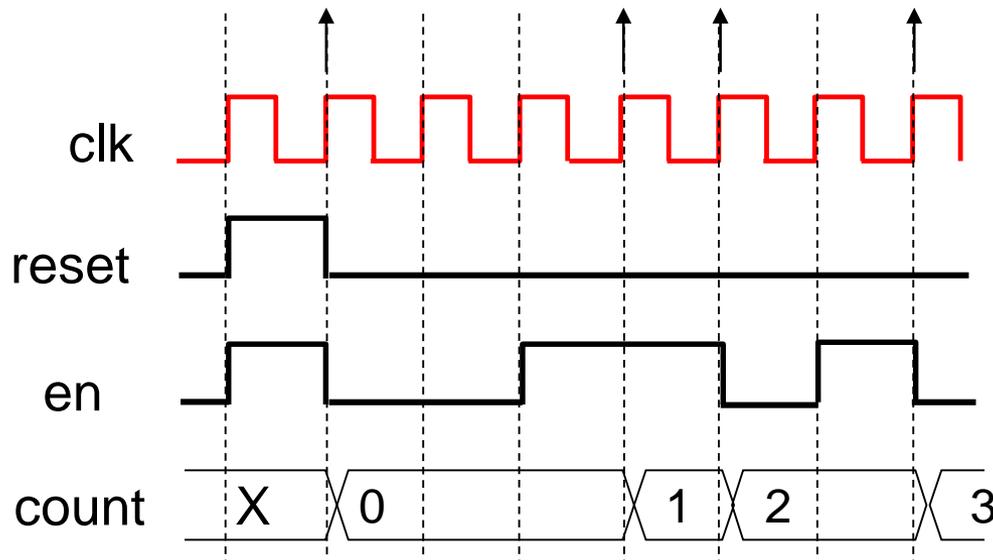


複数ビットをまとめて
このように描く

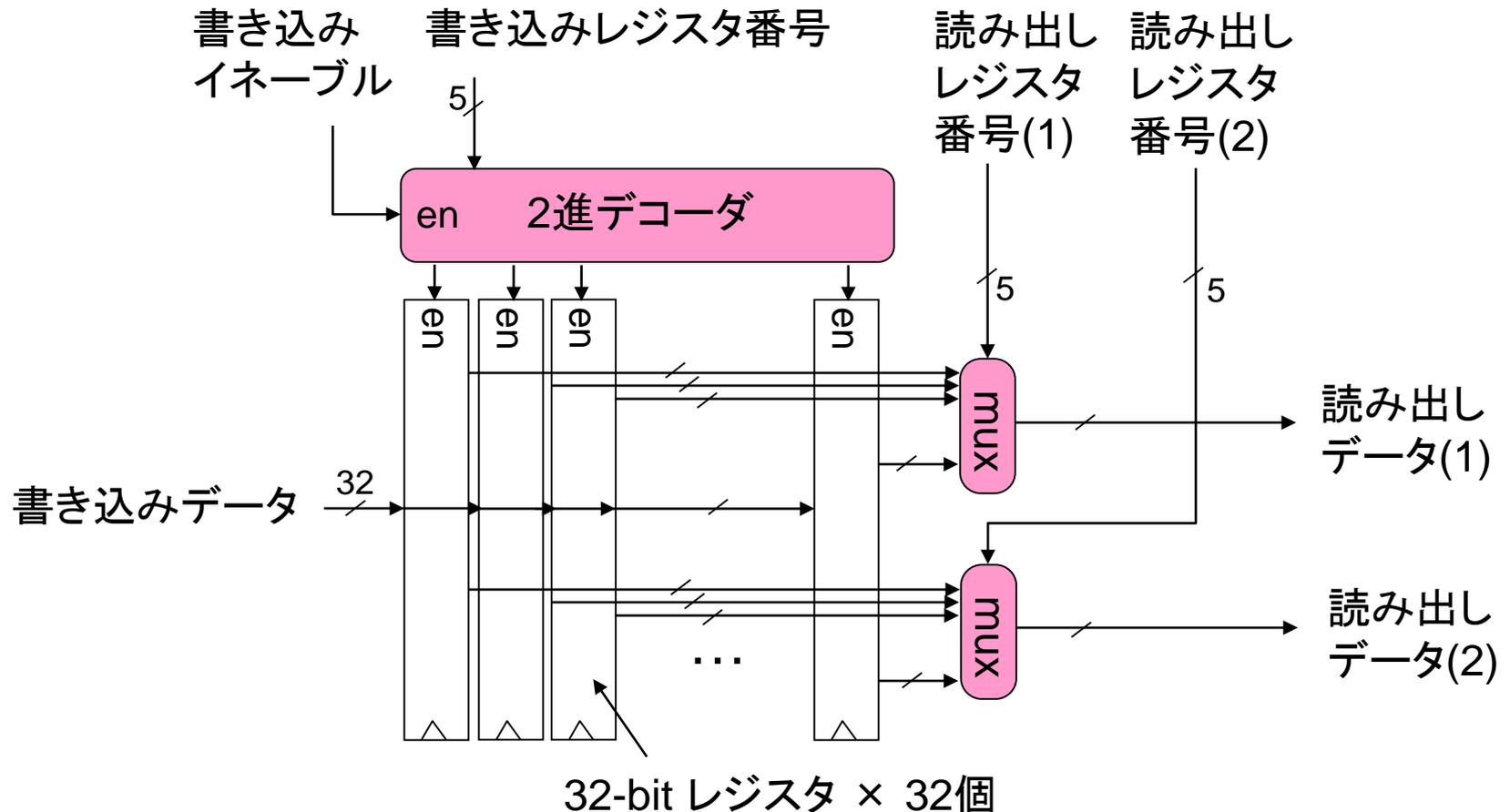
不定をこのように描いて
済ますこともある

例: 同期2進カウンタ

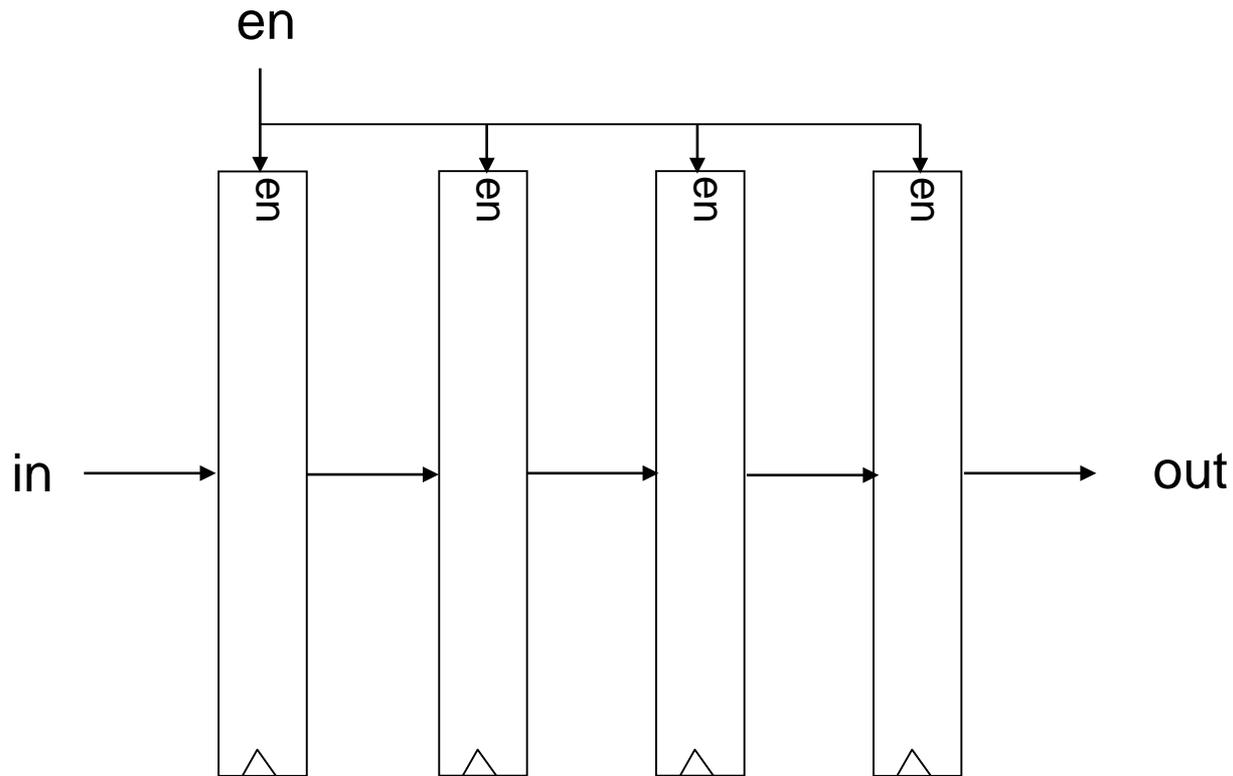
クロックの立上りの瞬間に en が1だった回数を数える



例: レジスタファイル (32 × 32ビット, 1入力2出力, 再掲)



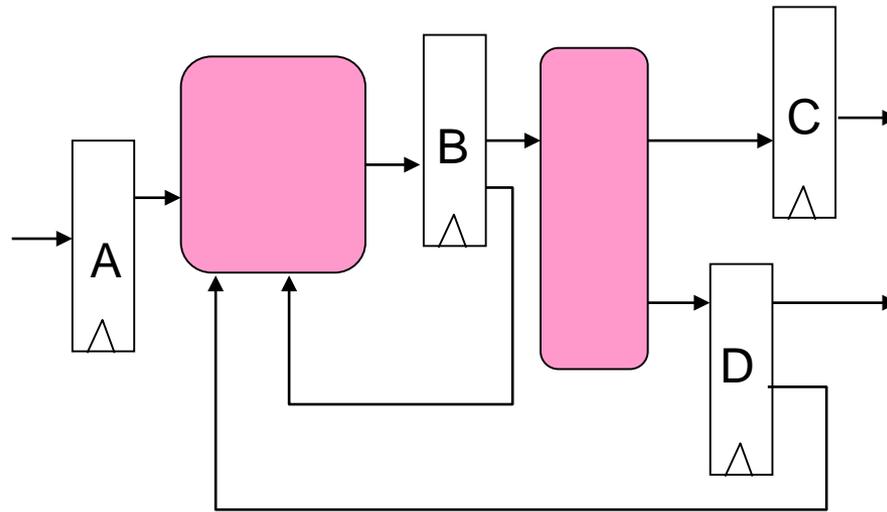
例: シフトレジスタ



FIFO (first-in first-out) の記憶回路 (queue) として用いられる

レジスタ間遅延とクリティカルパス

- 同期式順序回路: レジスタ間を組合せ回路でつないだもの
- クリティカルパス: レジスタ間の遅延が最大であるような信号経路



A → B: 8 ns

B → B: 5 ns

B → C: 2.5 ns

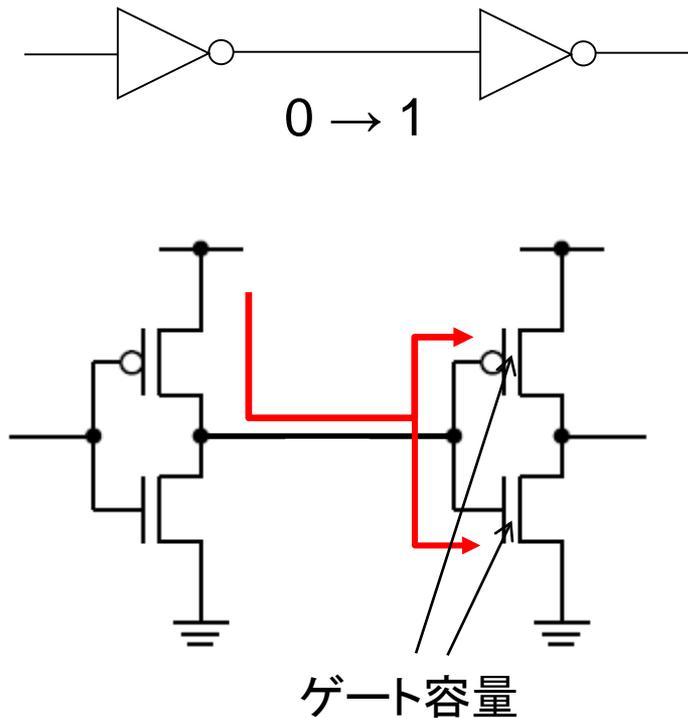
B → D: 7 ns

D → B: **12.5 ns**

クリティカルパスの遅延で
全体のクロック周波数の上限が決まる

$$1 / (12.5 \times 10^{-9}) = 80 \text{ MHz}$$

クロック周波数を制限するもの



ゲート遅延:

- トランジスタが流せる電流は有限
- したがって, ゲート容量を充電してトランジスタを反転させるには一定の時間が必要
- 理想的には, トランジスタサイズの微細化とともに小さくなる

配線遅延:

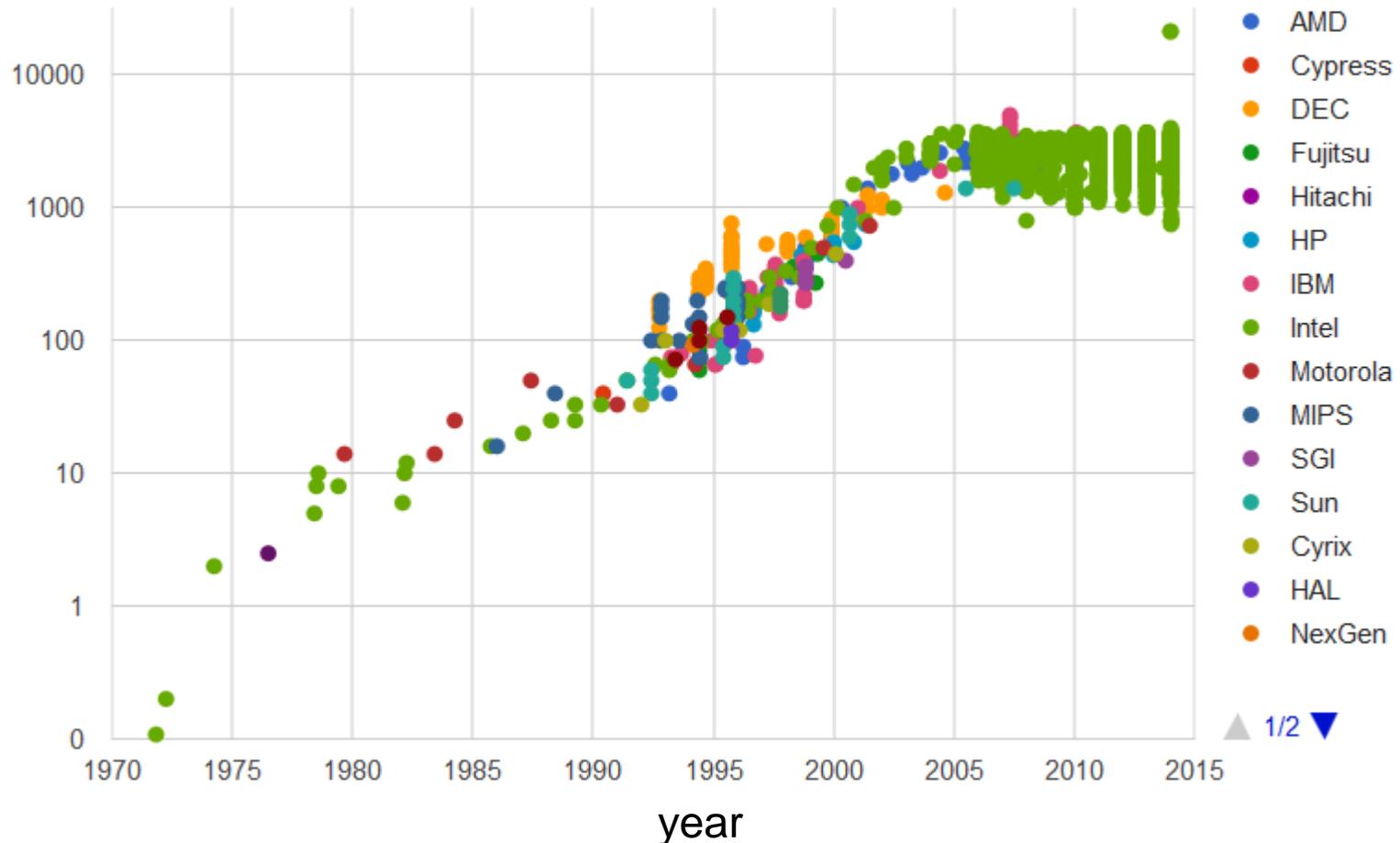
- そもそも光速は超えられない
(30万 [km/s] = 300 [mm/ns])
- 抵抗・容量・インダクタンスの影響

発熱:

- スイッチング = エネルギー消費
- 温度が上がりすぎると動作できない

プロセッサのクロック周波数の推移

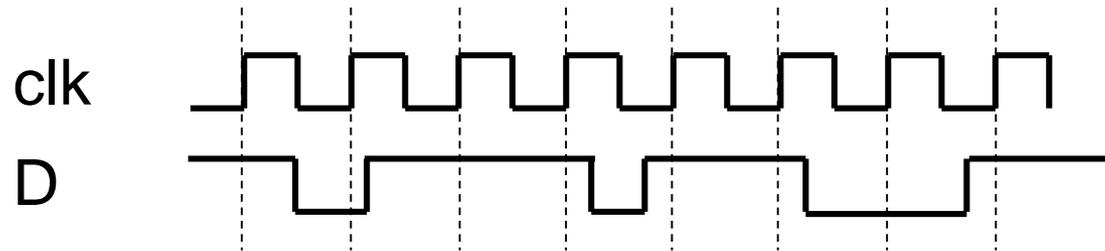
clock frequency (MHz)



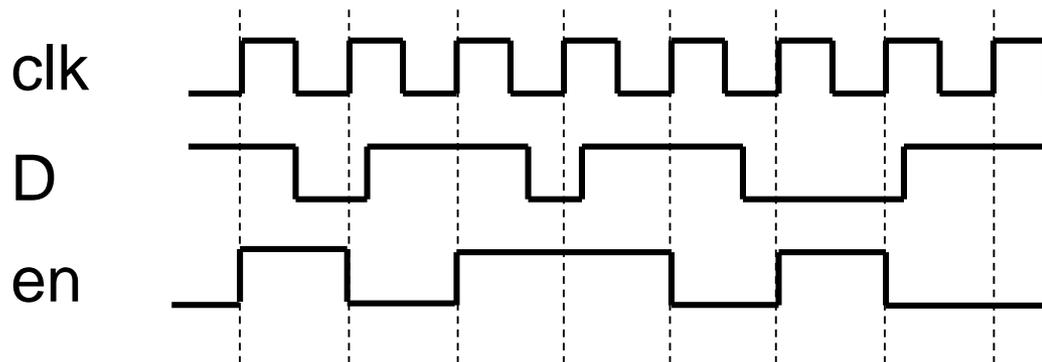
http://cpudb.stanford.edu/visualize/clock_frequency

練習問題(1)

1. 次のような入力を受け取る D-FF の状態(出力) Q はどのような信号になるか? ただし Q の初期状態は 0 とする.

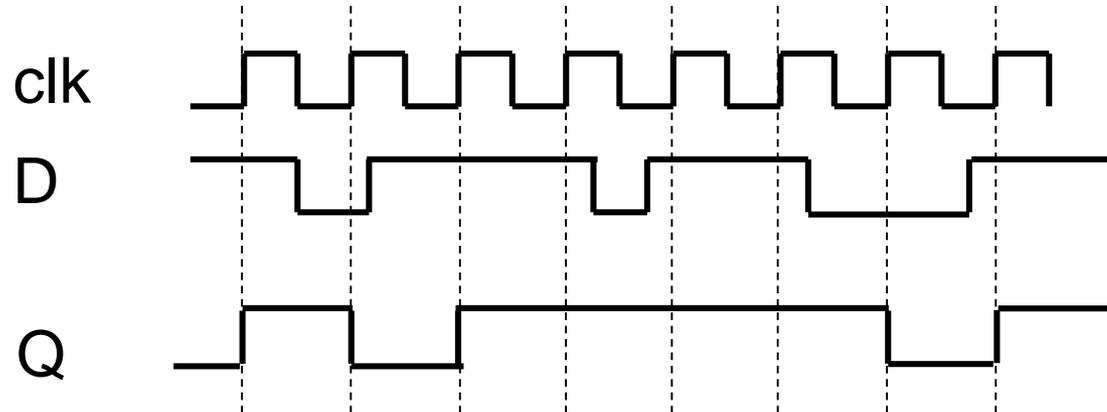


2. 次のような入力を受け取るイネーブル付き D-FF の状態 Q はどのような信号になるか? ただし Q の初期状態は 0 とする.

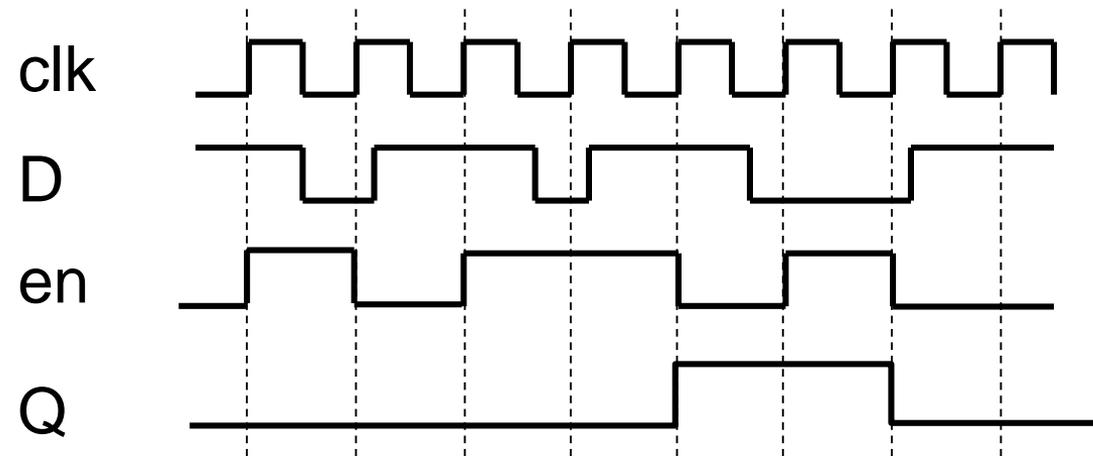


解答例

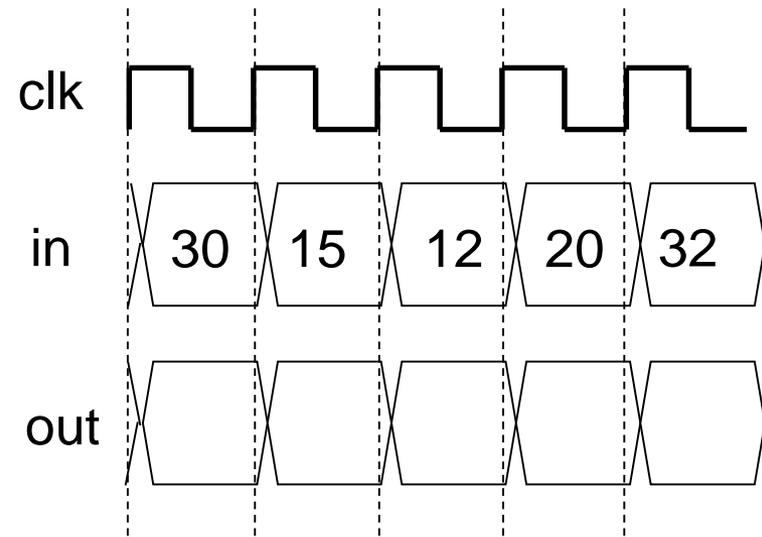
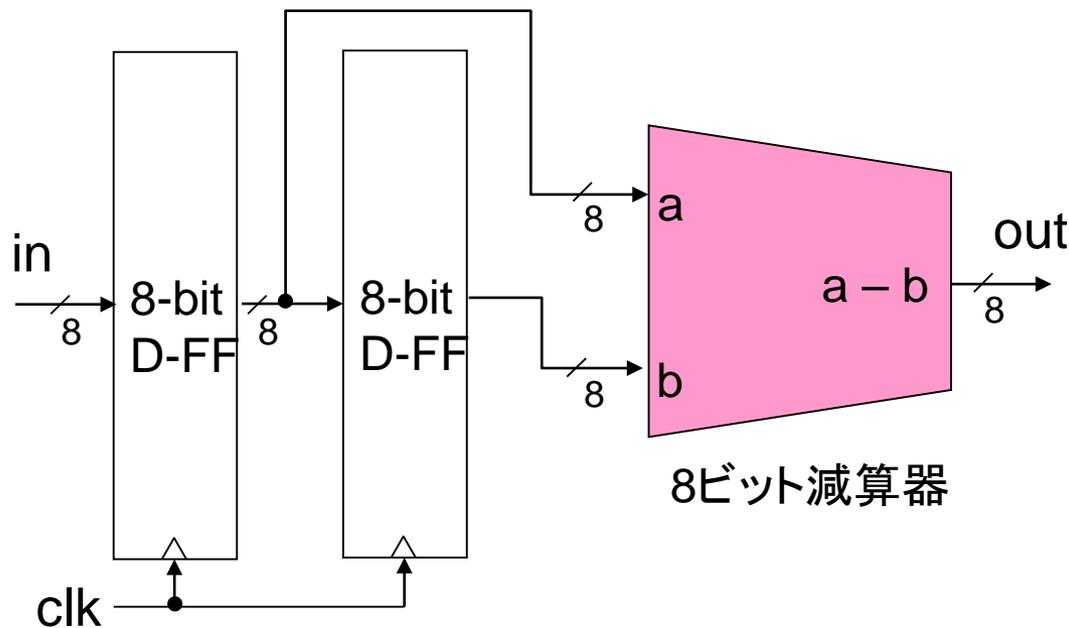
1.



2.



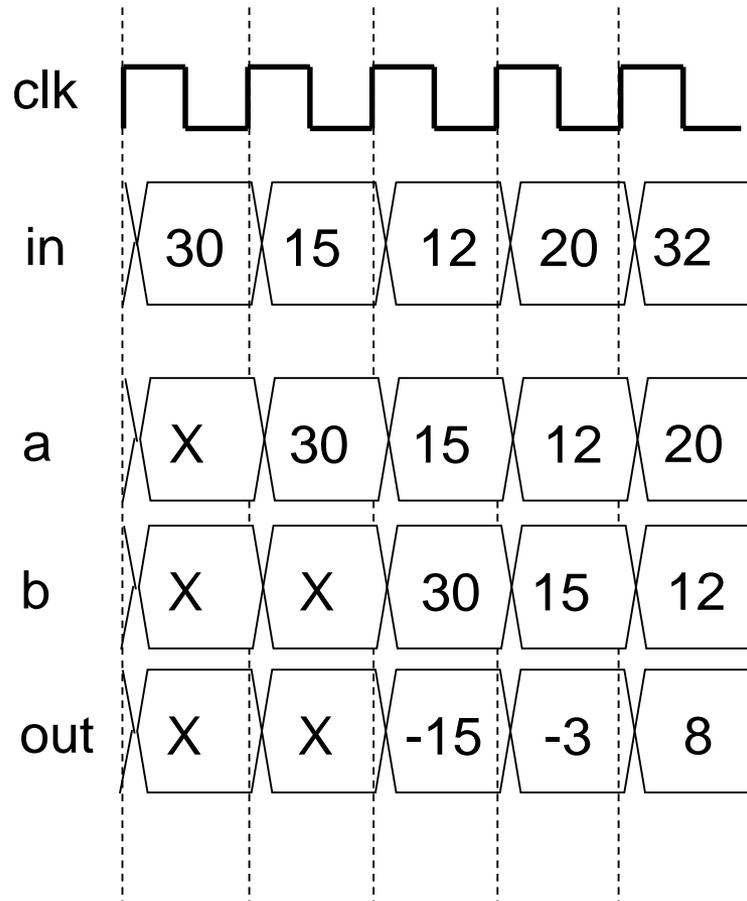
練習問題(2)



1. 入力信号 in を10進数で表示した際の値が右のタイミングチャートのように変化したとする. 出力信号 out はどのように変化するか. ただし, 値が不定であるときは × を記入すること.
2. 図中の8ビット減算器の代わりに8ビット加算器を置き, その直後に1ビット右シフト回路を置いたとする. この改造によって, 図の順序回路が計算するものは何から何に変化するか説明せよ. ただし, オーバフローは考慮しなくてよい.
(2012年度期末試験・改)

解答例

1.



2. 隣接時刻間の差分の代わりに、隣接時刻間の平均を計算することになる.