

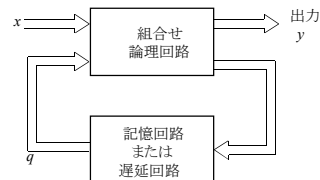
デジタル電子回路

順序回路

組み合わせ論理回路

出力の状態は入力の状態により決定される。
(記憶や前の状態により出力を変えることはできない。)
前の記録がない!

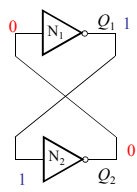
順序回路: 出力が現在の入力、**および過去の入力**によって決定。



フリップフロップ

過去の状態を記憶するための回路
→ 帰還作用を利用して1ビットの情報を記憶

2個のNOT回路の入出力を相互接続した回路を考えると...

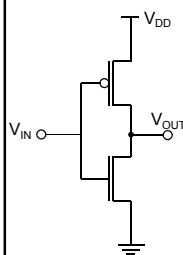


N_1 の出力 Q_1 が1だとすると
 N_2 に1が入力され Q_2 は0となる。これが再び
 N_1 に入力される。 Q_1 は1となり、回路は安定。

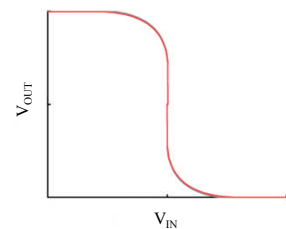
N_1 の出力 Q_1 が0だとすると
 N_2 に0が入力され Q_2 は1となる。これが再び
 N_1 に入力される。 Q_1 は0となり、回路は安定。

2安定回路 または **双安定回路**
→ **双安定マルチバイブレータ**とも呼ばれる

インバータの動作

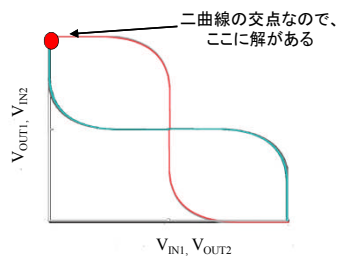
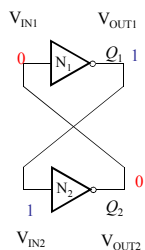


伝達特性(Transfer curve)



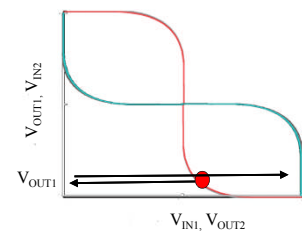
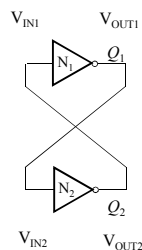
双安定の動作

二つのインバータをつなぎ合わせる



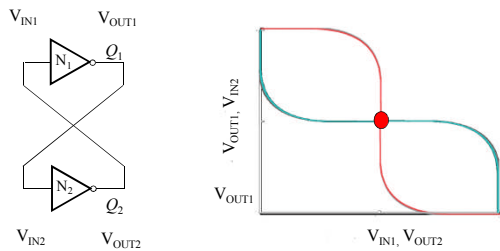
双安定の動作

V_{IN1} に強制的に $V_{DD}/2$ をすこし超える電圧をいれたら?



N_2 は V_{OUT2} にはほぼ V_{DD} の値を出力したい。
強制的な電圧が切れたら、 V_{DD} になる。

双安定の動作

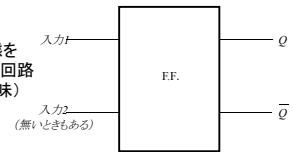


真ん中は非常に不安定。
数式的にはここで安定できるが、
雑音などで通常どっかに行く。

フリップフロップ (2)

フリップフロップ

2安定回路の二つの安定状態を
外部より選択できるようにした回路
(英語でバタバタするという意味)



安定状態を決定するための
入力と、互いに相補の関係
にある出力Q, $\bar{Q}$ からなる。

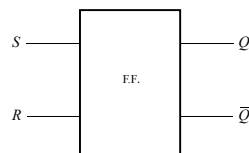
Q=1にすることを「フリップフロップを**セット**する」、

Q=0にすることを「フリップフロップを**リセット**する」という

SRフリップフロップ

S: セット入力
R: リセット入力

現在の出力 $Q^n, \bar{Q}^n$
↓
変化後の出力 $Q^{n+1}, \bar{Q}^{n+1}$



特性表

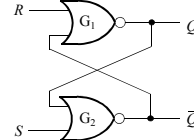
入 力	出力
S R	Q^{n+1}
0 0	Q^n
0 1	0
1 0	1
1 1	不定

- (1) S=0, R=0の時
出力は入力前の出力と同一
- (2) S=0, R=1の時
常に $Q^{n+1}=0$ (強制的にリセット)
- (3) S=1, R=0の時
常に $Q^{n+1}=1$ (強制的にセット)
- (4) S=1, R=1の時
禁止 (出力状態は定義されない。)

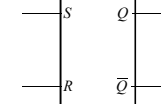
SRラッチ回路 SC(セット-クリア)フリップフロップ ということもある。

SRフリップフロップ (NOR構成)

注: Rが上 (出力Q) 側



記号



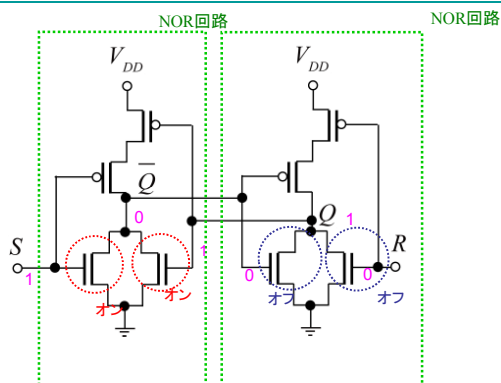
A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

特性表

入 力	出力
S R	Q^{n+1}
0 0	Q^n
0 1	0
1 0	1
1 1	不定

- (1) S=0, R=0の時
 $Q=1$ なら G_2 には1が入力され、 $\bar{Q}=0$ となる。
これが G_1 に入力されるので $Q=1$ で変化なし
 $Q=0$ なら G_2 には0が入力され、 $Q=1$ となる。
これが G_1 に入力されるので $Q=0$ で変化なし
- (2) S=0, R=1の時
 $R=1$ であるから $\bar{Q}$ にかかわらず $Q=0$
- (3) S=1, R=0の時
 $S=1$ であるから Q にかかわらず $\bar{Q}=0$ 、即ち $Q=1$
- (4) S=1, R=1の時
禁止 (両方0になり相補の出力にならない。)

SRフリップフロップ (NOR構成CMOS回路)

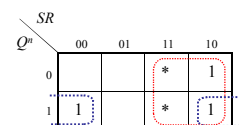


SRフリップフロップ (特性方程式)

S, R, Q^n を入力、 Q^{n+1} を出力として
真理値表を書くとき...

S	R	Q^n	Q^{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	*
1	1	1	*

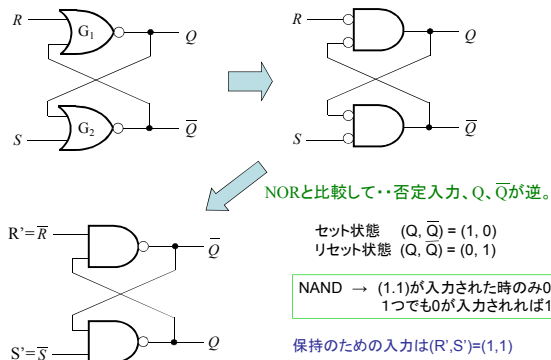
- (1) S=0, R=0の時
出力は入力前の出力と同一
- (2) S=0, R=1の時
常に $Q^{n+1}=0$ (強制的にリセット)
- (3) S=1, R=0の時
常に $Q^{n+1}=1$ (強制的にセット)
- (4) S=1, R=1の時
禁止 (出力状態は定義されない。)



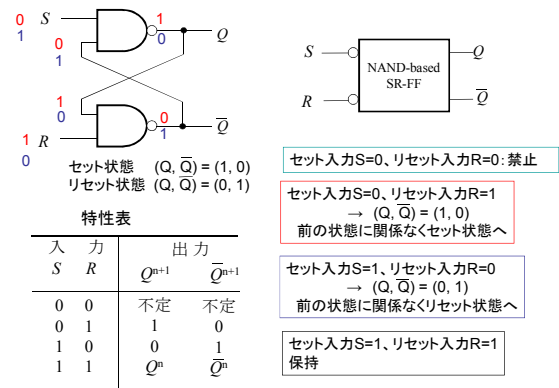
$$Q^{n+1} = S + Q^n \bar{R}$$

これを特性方程式という。

SRラッチ (NAND構成)



SRフリップフロップ (NAND構成) の動作

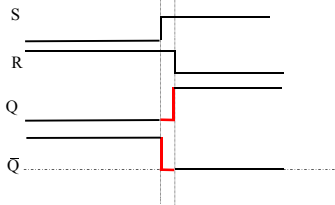


非同期形

入力に変化すれば状態が直ちに变化する。 **非同期形**

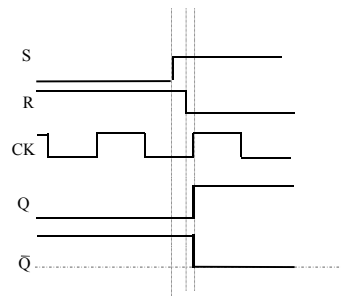
たとえば $(S, R) = (0, 1) \rightarrow (S, R) = (1, 0)$ と变化する場合、R入力の立ち下がりがS入力の立ち上がりより遅れると、 $(S, R) = (1, 1)$ が入力され、誤動作の可能性がある。

ハザード: ゲートの伝搬遅延などにより信号のタイミングにずれが生じ、本来の論理演算では想定していない動作が生じること。

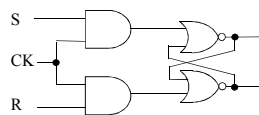


ハザードを防ぐには→同期形

同期形 クロックが入ったときだけ变化する。



同期形SRフリップフロップ

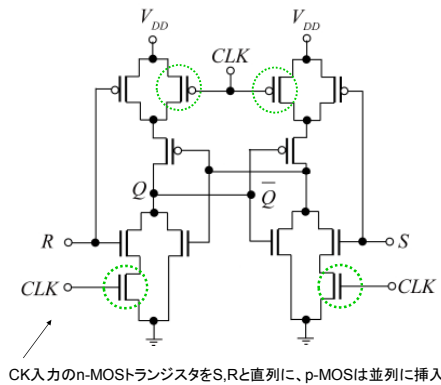


(注) $S=R=1$ は禁止

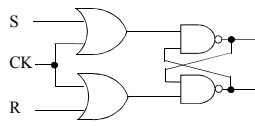
S、R入力とCKのANDをとる。

CK=1の時のみ動作
CK=0の場合は保持

同期形SRフリップフロップ (CMOS回路)



NANDを用いた構成

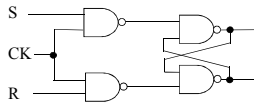


S, R入力とCKのORをとる。

CK=1の時は保持
CK=0の場合に動作

Active Low

クロック信号 CK=1の時に動作するようにすることも可能

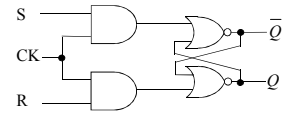


S, RもActive High (正論理)

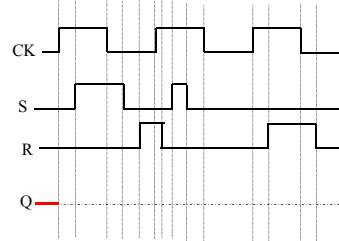
S=1, R=0 でセット(Q, $\bar{Q}$)=(1, 0)
S=0, R=1 でリセット(Q, $\bar{Q}$)=(0, 1)

例題

図に示すようなSRフリップフロップに
以下のようにクロック(CK)とS入力、
R入力が印加されたとする。
Qはどのように変化するか？

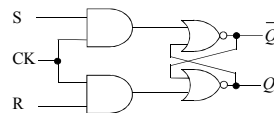


初期状態はQ=0とする

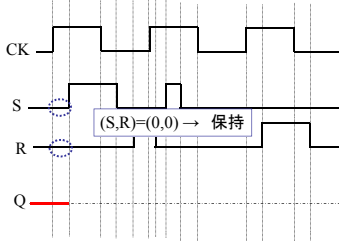


例題解答：同期型SRフリップフロップ

図に示すようなSRフリップフロップに
以下のようにクロック(CK)とS入力、
R入力が印加されたとする。
Qはどのように変化するか？
初期状態はQ=0とする。



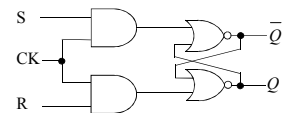
CK=1の時のみ動作



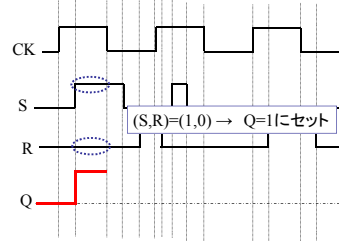
- (1) S=0, R=0の時 保持
- (2) S=0, R=1の時 リセット Q=0
- (3) S=1, R=0の時 セット Q=1
- (4) S=1, R=1の時 禁止

例題解答：同期型SRフリップフロップ

図に示すようなSRフリップフロップに
以下のようにクロック(CK)とS入力、
R入力が印加されたとする。
Qはどのように変化するか？
初期状態はQ=0



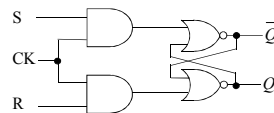
CK=1の時のみ動作



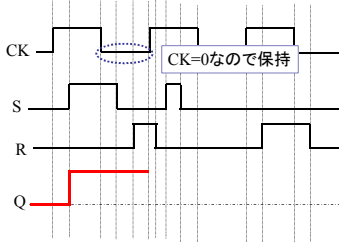
- (1) S=0, R=0の時 保持
- (2) S=0, R=1の時 リセット Q=0
- (3) S=1, R=0の時 セット Q=1
- (4) S=1, R=1の時 禁止

例題解答：同期型SRフリップフロップ

図に示すようなSRフリップフロップに
以下のようにクロック(CK)とS入力、
R入力が印加されたとする。
Qはどのように変化するか？
初期状態はQ=0



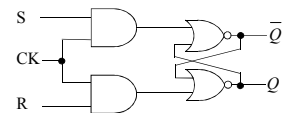
CK=1の時のみ動作



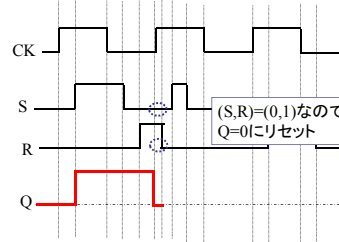
- (1) S=0, R=0の時 保持
- (2) S=0, R=1の時 リセット Q=0
- (3) S=1, R=0の時 セット Q=1
- (4) S=1, R=1の時 禁止

例題解答：同期型SRフリップフロップ

図に示すようなSRフリップフロップに
以下のようにクロック(CK)とS入力、
R入力が印加されたとする。
Qはどのように変化するか？
初期状態はQ=0



CK=1の時のみ動作

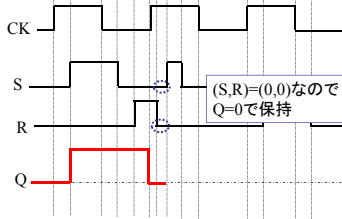


- (1) S=0, R=0の時 保持
- (2) S=0, R=1の時 リセット Q=0
- (3) S=1, R=0の時 セット Q=1
- (4) S=1, R=1の時 禁止

例題解答: 同期型SRフリップフロップ

図に示すようなSRフリップフロップに以下のように入力信号(CK)とS入力、R入力が印加されたとする。Qはどのように変化するか？初期状態はQ=0

CK=1の時のみ動作

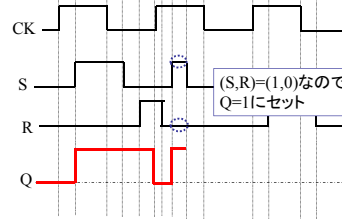


- (1) S=0, R=0の時 保持
- (2) S=0, R=1の時 リセット Q=0
- (3) S=1, R=0の時 セット Q=1
- (4) S=1, R=1の時 禁止

例題解答: 同期型SRフリップフロップ

図に示すようなSRフリップフロップに以下のように入力信号(CK)とS入力、R入力が印加されたとする。Qはどのように変化するか？初期状態はQ=0

CK=1の時のみ動作

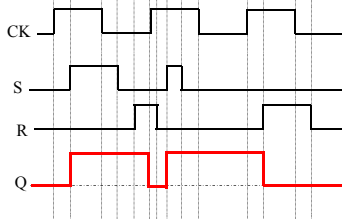


- (1) S=0, R=0の時 保持
- (2) S=0, R=1の時 リセット Q=0
- (3) S=1, R=0の時 セット Q=1
- (4) S=1, R=1の時 禁止

例題解答: 同期型SRフリップフロップ

図に示すようなSRフリップフロップに以下のように入力信号(CK)とS入力、R入力が印加されたとする。Qはどのように変化するか？初期状態はQ=0

CK=1の時のみ動作

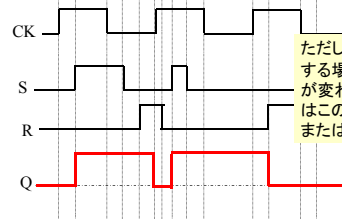


- (1) S=0, R=0の時 保持
- (2) S=0, R=1の時 リセット Q=0
- (3) S=1, R=0の時 セット Q=1
- (4) S=1, R=1の時 禁止

例題解答: 同期型SRフリップフロップ

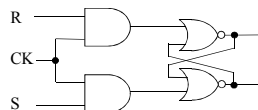
図に示すようなSRフリップフロップに以下のように入力信号(CK)とS入力、R入力が印加されたとする。Qはどのように変化するか？初期状態はQ=0

CK=1の時のみ動作



ただし、FFでカウンタやレジスタを構成する場合、クロックパルス中に入力信号が変わると不具合が起こるので、普通はこのような入力パターンは避ける。またはエッジトリガ型のFFを用いる。

同期型SRフリップフロップ



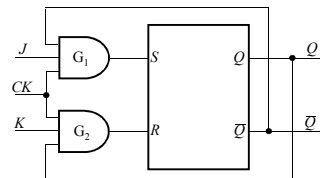
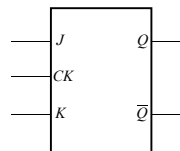
動作の表

S	R	Q^n	$\bar{Q}^n$	S'	R'	Q^{n+1}	$\bar{Q}^{n+1}$
0	0	0	1	0	0	0	1
		1	0	0	0	1	0
0	1	0	1	0	1	0	1
		1	0	0	1	0	1
1	0	0	1	1	0	1	0
		1	0	1	0	1	0
1	1	0	1	1	1	0	0
		1	0	1	1	0	0

JKフリップフロップ

SRフリップフロップ: 禁止入力の組み合わせがある。→ JKフリップフロップ

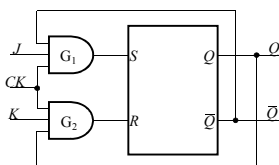
CK=0の時は保持
CK=1の場合に動作



- (1) J=0, K=0の時 出力は入力前の出力と同一(保持)
- (2) J=0, K=1の時 常に $Q^{n+1}=0$ (強制的にリセット)
- (3) J=1, K=0の時 常に $Q^{n+1}=1$ (強制的にセット)
- (4) J=1, K=1の時 $Q^{n+1} = \bar{Q}^n$ (反転)

入 力		出力
J	K	Q^{n+1}
0	0	Q^n
0	1	0
1	0	1
1	1	$\overline{Q^n}$

JKフリップフロップ



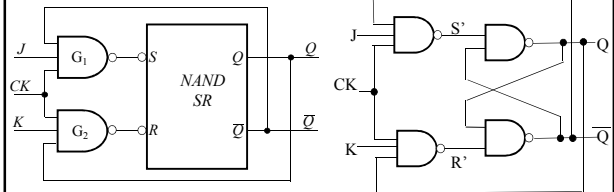
入	力	出力
J	K	Q^{n+1}
0	0	Q^n
0	1	0
1	0	1
1	1	$\bar{Q}^n$

- (1) $J=0, K=0$ の時
 G_1, G_2 はANDゲートなので、 $Q, \bar{Q}$ にかかわらず、 $S=0, R=0$ で保持。
- (2) $J=0, K=1$ の時
 $S=0, Q=1$ の時は $R=1$ となりリセット。
 $Q=0$ の時は $R=0$ で保持。つまり常に $Q^{n+1}=0$ (強制的にリセット)
- (3) $J=1, K=0$ の時
 $R=0, Q=1$ ($\bar{Q}=0$)の時は $S=0$ となり保持。
 $Q=0$ ($\bar{Q}=1$)の時の時は $S=1$ となり、セット。つまり...常に $Q^{n+1}=1$ (強制的にセット)
- (4) $J=1, K=1$ の時
 - $Q=1$ ($\bar{Q}=0$)の時は $(S, R)=(0, 1)$ となりリセットされ、 $Q=0$ と反転する。
 - $Q=0$ ($\bar{Q}=1$)の時は $(S, R)=(1, 0)$ となりセットされ、 $Q=1$ と反転する。 $Q^{n+1} = \bar{Q}^n$ (反転)

JKフリップフロップ (NAND構成)

NANDでの構成

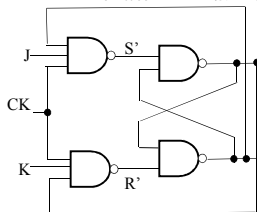
CK=0の時は保持
CK=1の場合に動作



問題

NAND構成のJKフリップフロップについて、以下の表を完成させてみよう。

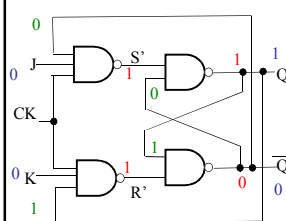
CK=0の時は保持 CK=1の場合に動作



J	K	Q^n	$\bar{Q}^n$	S'	R'	Q^{n+1}	$\bar{Q}^{n+1}$
0	0	0	1	1	1	0	1
		1	0				
0	1	0	1				
		1	0				
1	0	0	1				
		1	0				
1	1	0	1				
		1	0				

NAND構成JKフリップフロップ

NAND構成のJKフリップフロップについて、以下の表を完成させよ

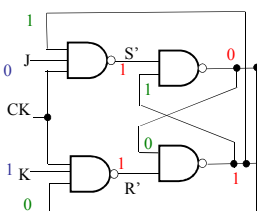


A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0

J	K	Q^n	$\bar{Q}^n$	S'	R'	Q^{n+1}	$\bar{Q}^{n+1}$
0	0	0	1	1	1	0	1
		1	0	1	1	1	0
0	1	0	1				
		1	0				
1	0	0	1				
		1	0				
1	1	0	1				
		1	0				

NAND構成JKフリップフロップ

NAND構成のJKフリップフロップについて、以下の表を完成させよ

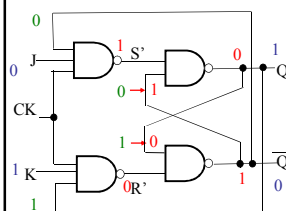


A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0

J	K	Q^n	$\bar{Q}^n$	S'	R'	Q^{n+1}	$\bar{Q}^{n+1}$
0	0	0	1	1	1	0	1
		1	0	1	1	1	0
0	1	0	1	1	1	0	1
		1	0				
1	0	0	1				
		1	0				
1	1	0	1				
		1	0				

NAND構成JKフリップフロップ

NAND構成のJKフリップフロップについて、以下の表を完成させよ

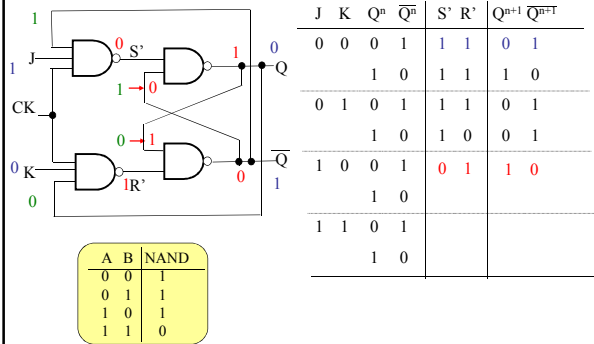


A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0

J	K	Q^n	$\bar{Q}^n$	S'	R'	Q^{n+1}	$\bar{Q}^{n+1}$
0	0	0	1	1	1	0	1
		1	0	1	1	1	0
0	1	0	1	1	1	0	1
		1	0	1	0	0	1
1	0	0	1				
		1	0				
1	1	0	1				
		1	0				

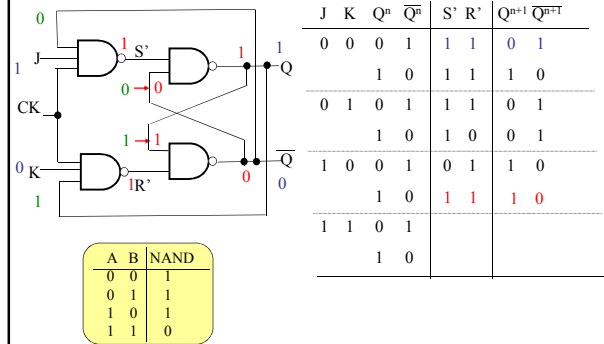
NAND構成JKフリップフロップ

NAND構成のJKフリップフロップについて、以下の表を完成させよ



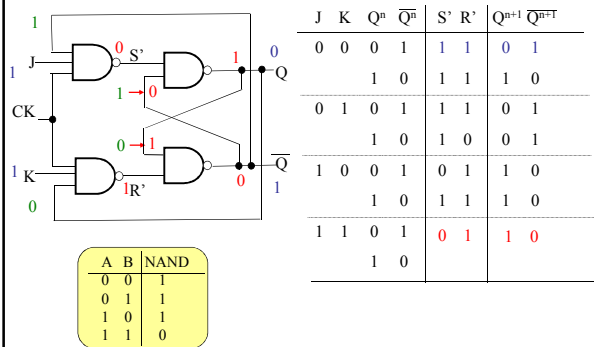
NAND構成JKフリップフロップ

NAND構成のJKフリップフロップについて、以下の表を完成させよ



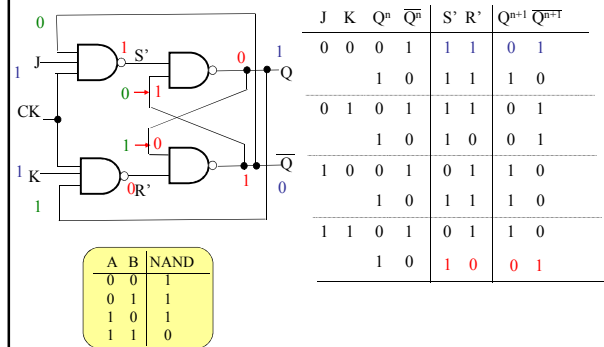
NAND構成JKフリップフロップ

NAND構成のJKフリップフロップについて、以下の表を完成させよ



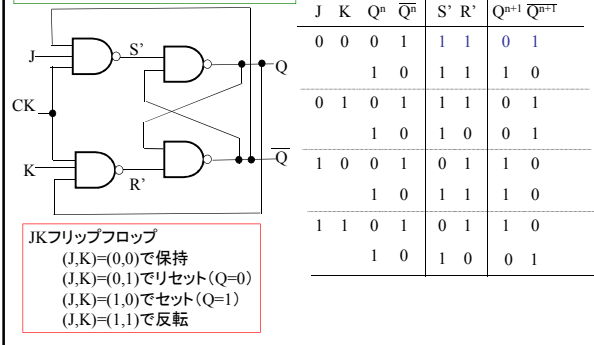
NAND構成JKフリップフロップ

NAND構成のJKフリップフロップについて、以下の表を完成させよ



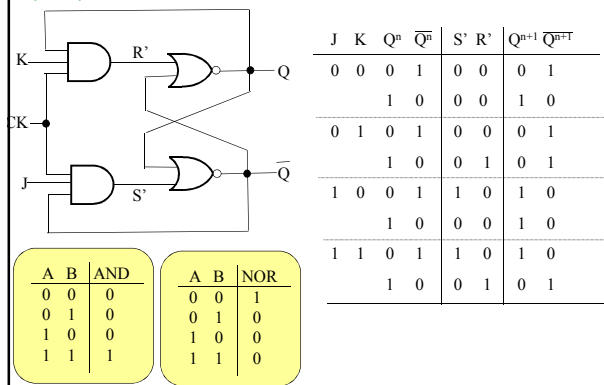
NAND構成JKフリップフロップ

NAND構成のSR-FF
 $S'=0$ でセット $R'=0$ でリセット
 $S'=1, R'=1$ は保持



JKフリップフロップ (NOR構成)

J,Kと S',R' の上下がNAND形と逆になっていることに注意!

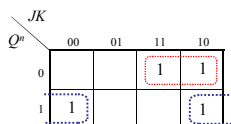


JKフリップフロップの特性方程式

J, K, Q^n を入力、 Q^{n+1} を出力として
真値表を書くと・・・

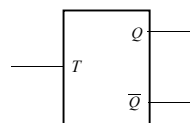
J	K	Q^n	Q^{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

- (1) J=0, K=0の時
出力は入力前の出力と同一
- (2) J=0, K=1の時
常に $Q^{n+1}=0$ (強制的にリセット)
- (3) J=1, K=0の時
常に $Q^{n+1}=1$ (強制的にセット)
- (4) J=1, K=1の時
 $Q^{n+1}=\overline{Q^n}$ (反転)



$$Q^{n+1} = J \cdot \overline{Q^n} + \overline{K} \cdot Q^n$$

Tフリップフロップ



Tが1になるごとに出力の
状態が反転する。
Toggle = ひっくり返る

JKフリップフロップのJ,K入力を1とし、CK入力をTとすれば実現できる。

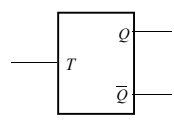
◎ T入力にパルスが2個印加されるごとに元の状態に戻る。

→ 2進のパルスカウンタ (従続につなげば、2進カウンタ)
Frequency divider (周波数分割器)

入力を出力に繋ぐと発振器になる 発振周波数はトグル周波数と呼ばれ、高速性の指標

Tフリップフロップにセット入力端子S、リセット入力端子Rを付加した
SR-Tフリップフロップというものもある。

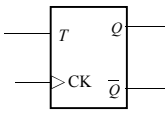
Tフリップフロップの構成例



非同期形

T	Q^{n+1}
0	Q^n
1	$\overline{Q^n}$

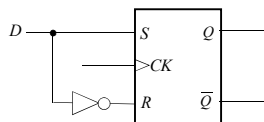
Tが1になるごとに出力の
状態が反転する。



同期形

Dラッチ

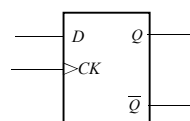
CK=1で入力を記憶、CK=0で出力状態を保持 (Dラッチ)



Dラッチは、現在の回路において、もっともよく使われているフリップフロップ
→
トランジションゲートという概念の導入により
マルチプレクサとNOTループ構造をもちいたD-FFが非常にコンパクトになるから

なお、一般的にはDだけでなく、セット入力端子S、リセット入力端子Rを付加して
強制的に出力が変えられるようになっているICが多い。

Dフリップフロップ

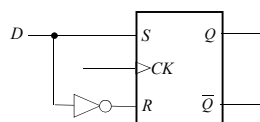


CK=1の時、入力と同一の出力。

入力Dの状態を一時的に記憶。
入力Dの情報をクロックパルスが入力されるまで遅らせる。(Delay)

(データフリップフロップ、遅延フリップフロップ)

CK=1で入力を記憶、CK=0で出力状態を保持 (Dラッチ)



フリップフロップの応用(レジスタ)

フリップフロップ: 情報の一時記憶 → レジスタとカウンタ

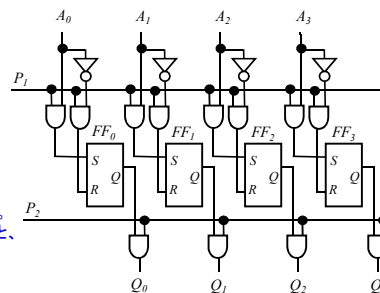
1個のフリップフロップ
1ビットの2進データを記憶。

n個のフリップフロップ
nビットの2進データを記憶。
(メモリレジスタ, orレジスタ)

2進データを $A_0 \sim A_3$ へ入力

P_1 : 書き込み信号

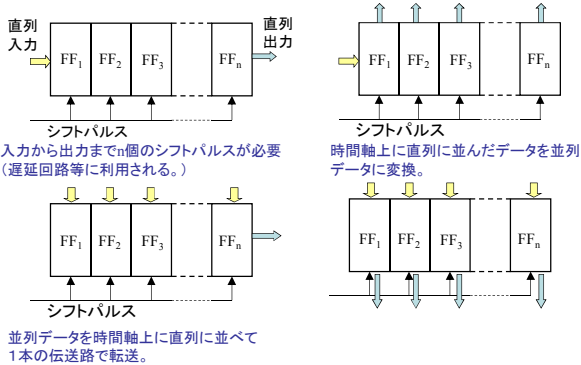
$P_1=1$ で $A_0 \sim A_3$ の状態が、
 $FF_0 \sim FF_3$ の各出力Qに記憶。
書き込み終了後 $P_1=0$ とすると、
FFのSR入力がともに0となり
状態を保持する。



並列入力並列出力レジスタ

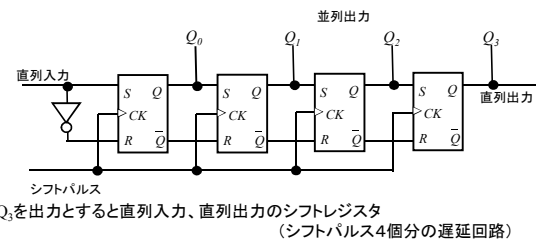
シフトレジスタ(1)

外部より加えられる信号(シフトパルス)により、各段のフリップフロップの内容が隣のフリップフロップに順次転送される。



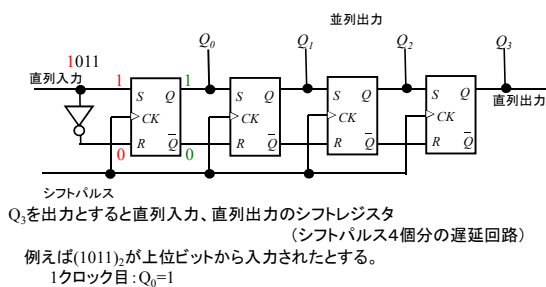
シフトレジスタ(2)

4ビット直列入力並列出力形シフトレジスタ。



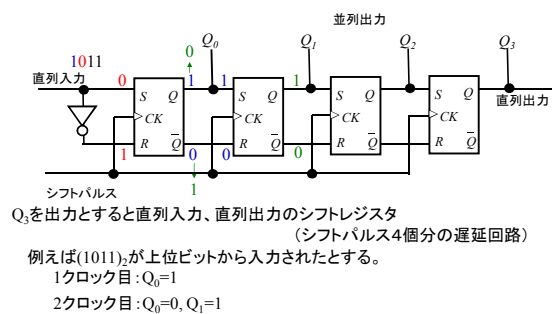
シフトレジスタ(2)

4ビット直列入力並列出力形シフトレジスタ。



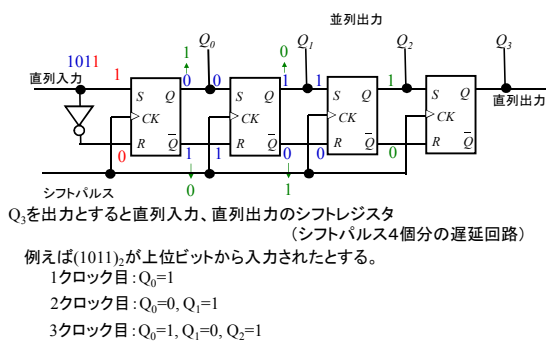
シフトレジスタ(2)

4ビット直列入力並列出力形シフトレジスタ。



シフトレジスタ(2)

4ビット直列入力並列出力形シフトレジスタ。



シフトレジスタ(2)

4ビット直列入力並列出力形シフトレジスタ。

