

コンピュータとネットワーク

第9回 論理回路

石井 健太郎

1308研究室・kenta@isc.senshu-u.ac.jp

スケジュール

- 4月11日 第1回「イントロダクション」
- 4月18日 第2回「コンピュータとその利用～ビジネスと情報システム」
- 4月25日 第3回「コンピュータ小史～情報の表現」
- 5月9日 第4回「コンピュータ小史～情報の表現」
- 5月16日 第5回「情報の表現」
- 5月23日 第6回「文字コードと日本語処理～中央処理装置」
- 5月30日 第7回「中央処理装置～記憶装置と入出力装置」
- 6月6日 第8回「記憶装置と入出力装置～論理回路」

スケジュール

6月13日 第9回「論理回路」

6月20日 第10回「**中間テスト**」

6月27日 第11回「テストのフィードバック、ソフトウェアの分類～
OSとプログラム言語～グラフによる表現」

7月4日 第12回「通信ネットワーク」

7月11日 第13回「インターネットとTCP/IP」

7月18日 第14回「セキュリティ」

7月25日 第15回「まとめと**授業内テスト**」

質問受け付け中！

- 予習時・授業中にわからなかったこと・質問したいことを、
随時responで入力してください
 - なるべく拾って答えるようにします
- もちろん、手を挙げて質問してもOK！

論理回路(つづき)

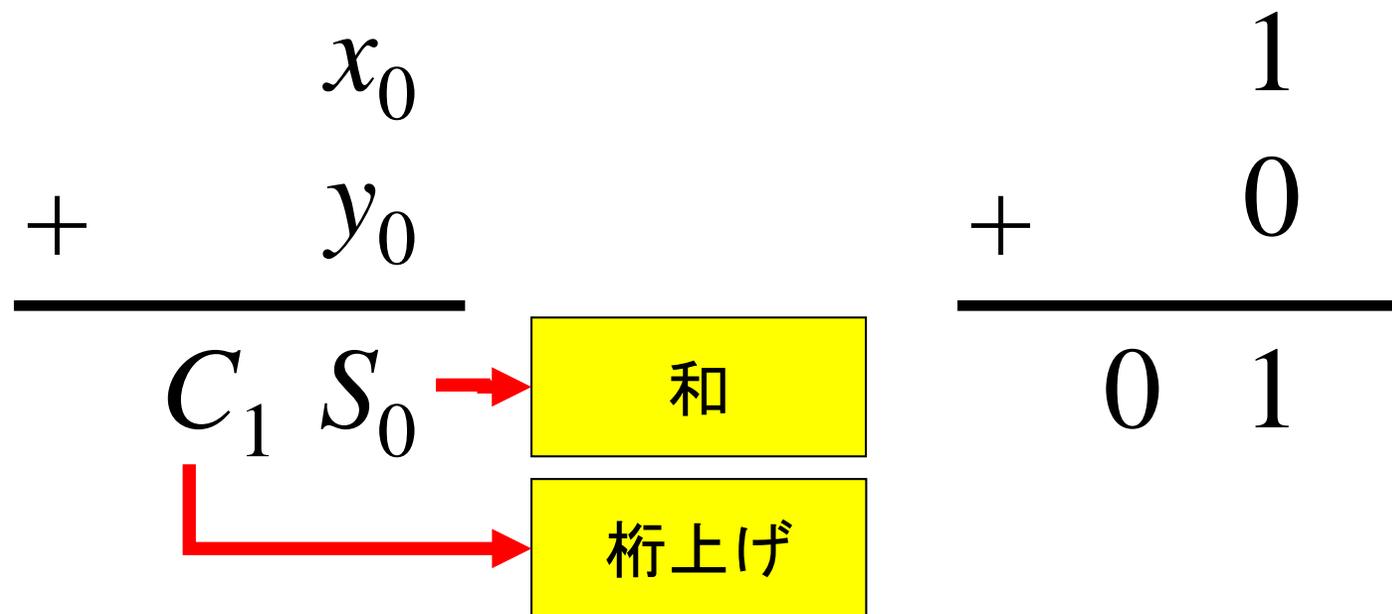
p.130~, p.138~

論理回路の作成

1. 機能から真理値表の作成
2. 主加法標準形論理式の作成
 - a. 真理値表で出力が1になる入力パターンに注目する
 - b. それらの入力パターンの論理積が1になるように項を作る
 - c. それらの項を論理和で結ぶ
3. 論理式から論理回路を作成
4. ド・モルガンの定理による単純化
(\Rightarrow NOTとNANDによる実装)

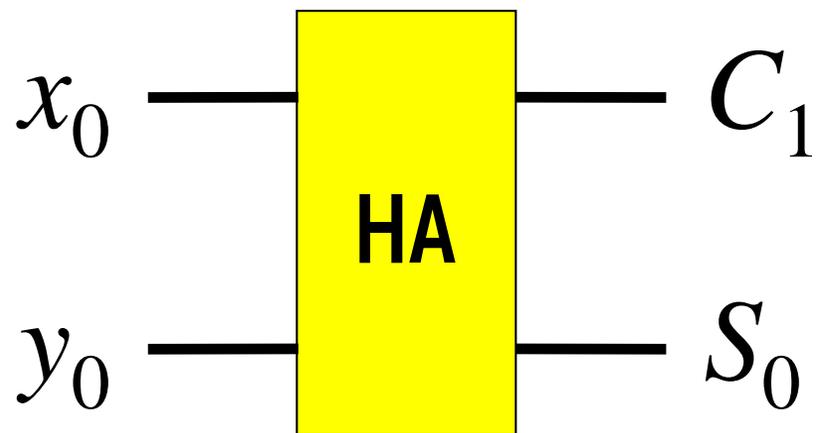
半加算器 (Half Adder)

- 1ビット同士の加算を行う回路



半加算器の真理値表

論理記号

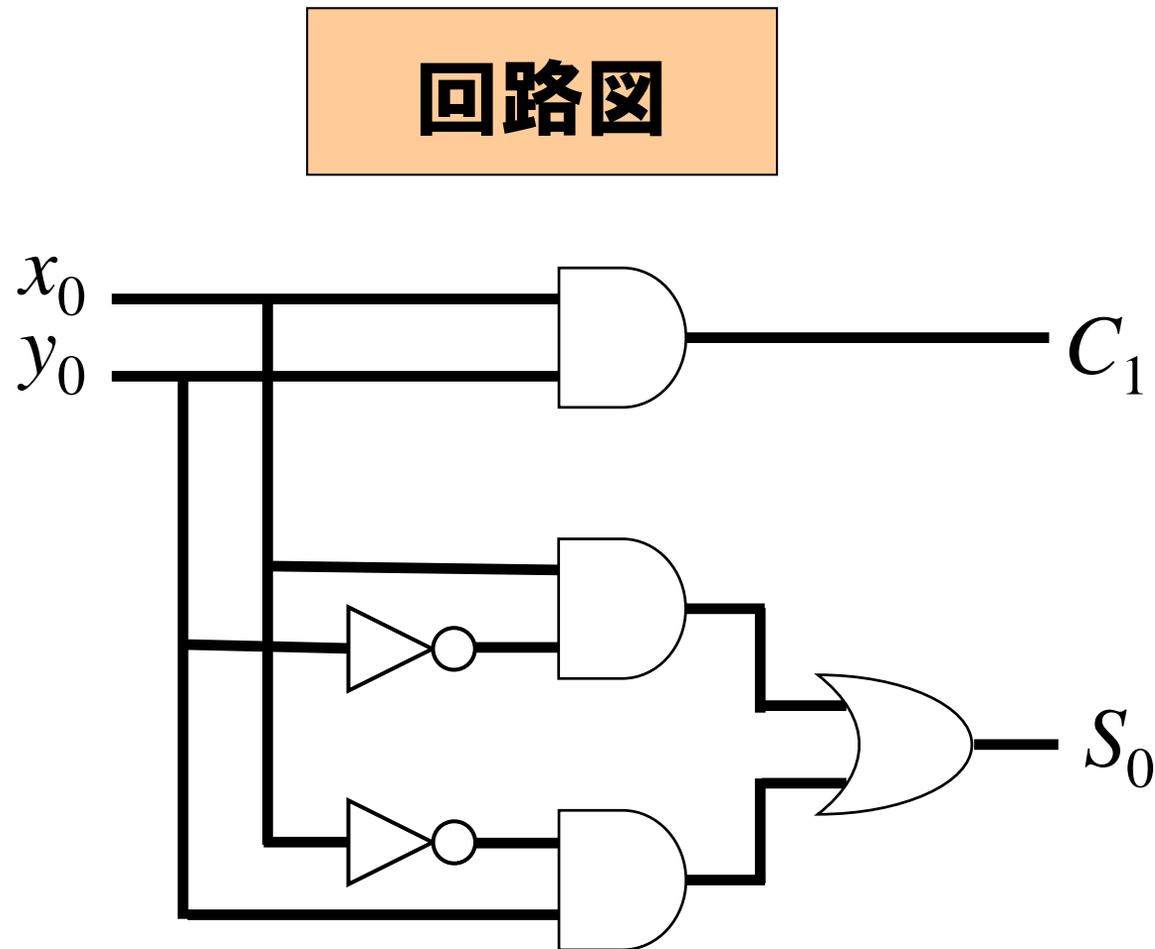


真理値表

x_0	y_0	S_0	C_1
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

半加算器の回路図

回路図

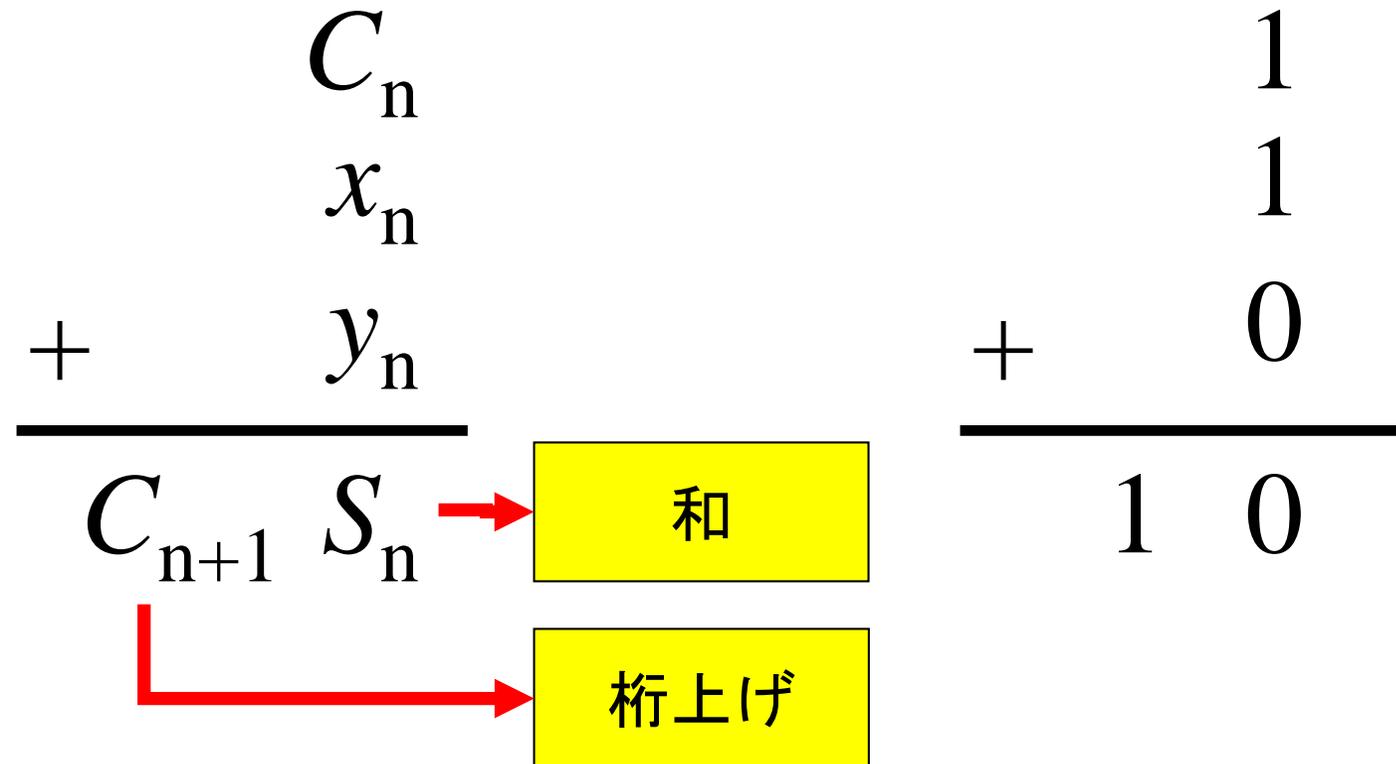


真理値表

x_0	y_0	S_0	C_1
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

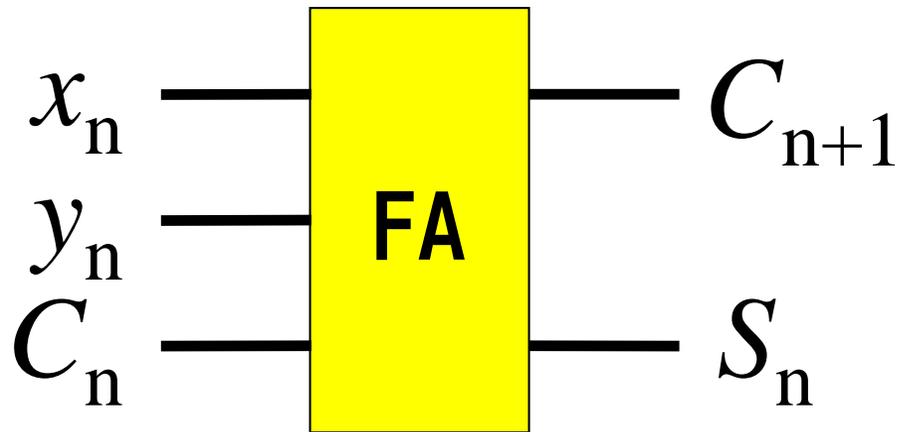
全加算器 (Full Adder)

- 桁上がりを考慮して、1ビット同士の加算を行う回路



全加算器の真理値表

論理記号



真理値表

x_n	y_n	C_n	S_n	C_{n+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

全加算器の回路図

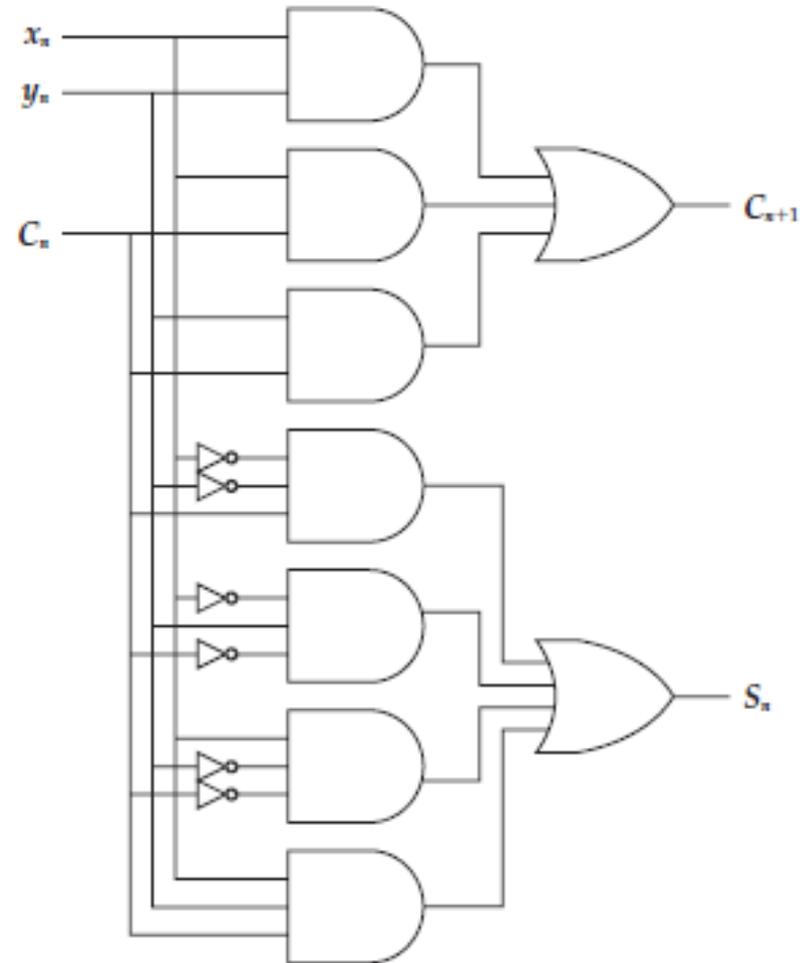


図 5.43 全加算器の論理回路図

4ビット加算器の論理回路図

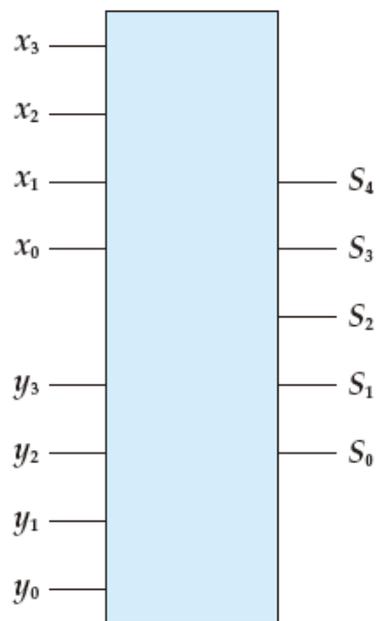


図 5.55 4ビット加算器のブロック図

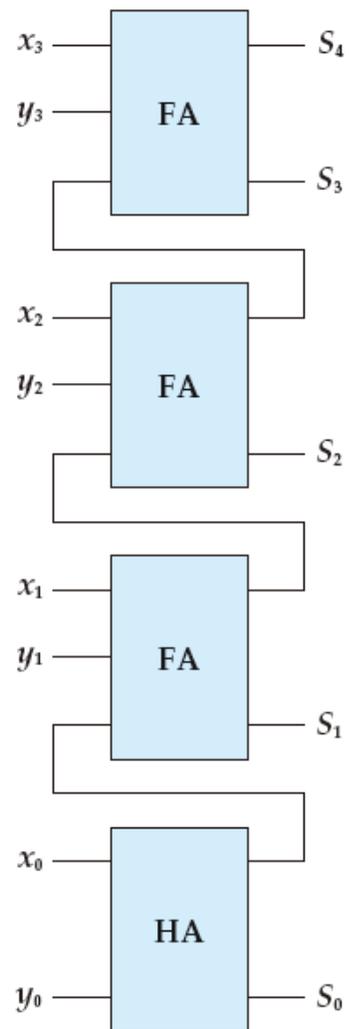
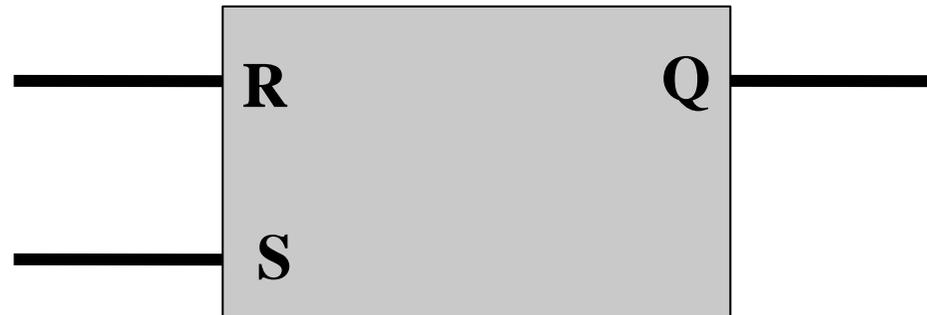


図 5.56 4ビット加算器の論理回路図

SRAM

- Static RAM
- RSフリップフロップ
 - R・・・リセット信号(記憶Qを0にする)
 - S・・・セット信号(記憶Qを1にする)
 - Q・・・記憶する1ビットのデータ

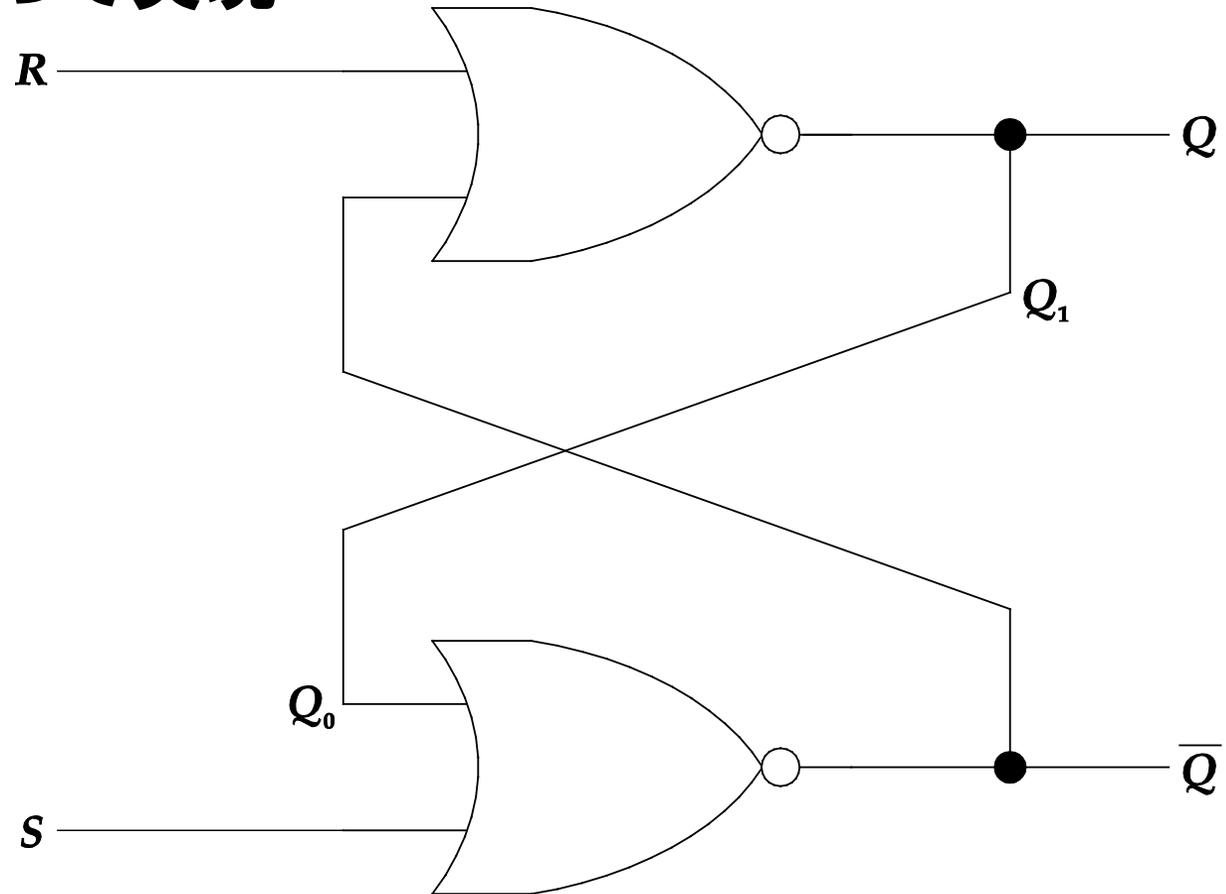


フリップフロップの真理値表

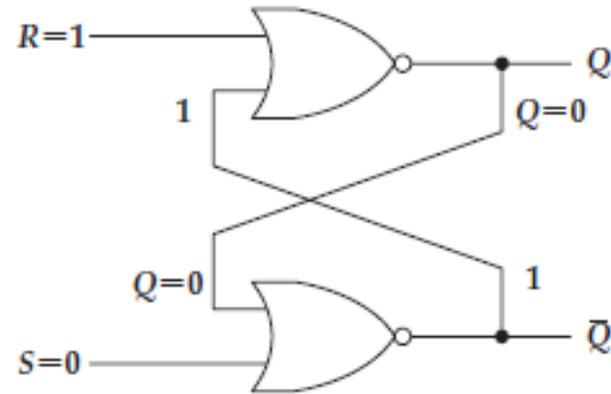
入力			出力
R	S	Q_0	Q_1
0	0	0	0(そのまま保持)
0	0	1	1(そのまま保持)
0	1	0	1(セット)
0	1	1	1(セット)
1	0	0	0(リセット)
1	0	1	0(リセット)

フリップフロップの論理回路

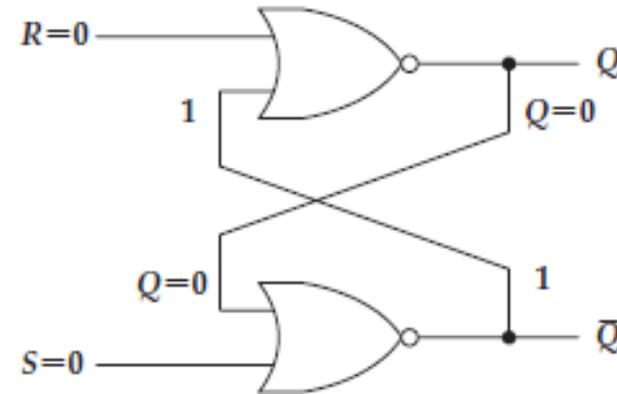
- NORゲート2つで実現



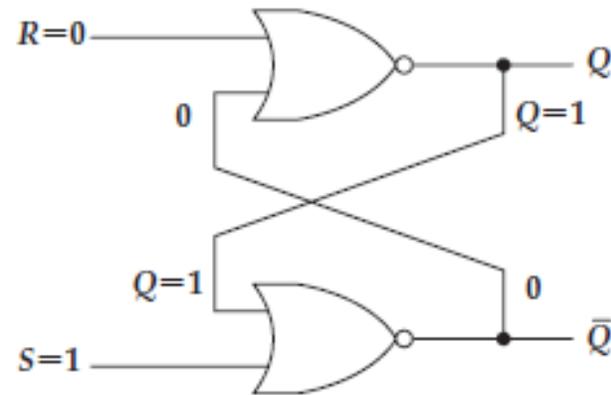
フリップフロップの動作



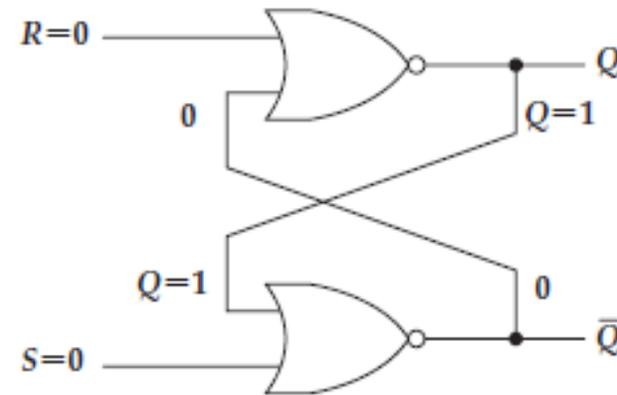
① $R=1, S=0$ のとき $Q=0$ にリセット



② $R=0, S=0$ のとき $Q=0$ を保持



③ $R=0, S=1$ のとき $Q=1$ にリセット



④ $R=0, S=0$ のとき $Q=1$ を保持

図 5.48 フリップフロップの動作遷移