

論理演算応用

半加算器

① 半加算器は、

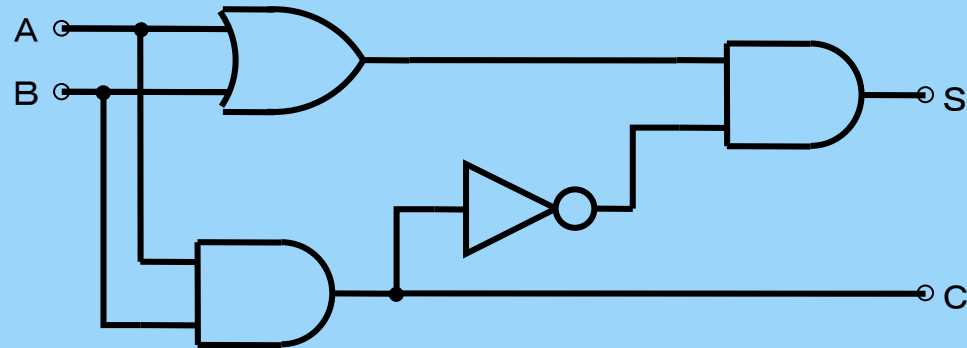
- ① 2個の2進数の和を求める。
- ② 桁上がりを考慮に入れない加算器である。
- ③ 1ビットの2進数の加算しかできない加算器である。

② 排他的論理和と論理積の
組み合わせで実現している。

③ 次の図において、

- ① Sは1桁の加算で、排他的論理和の結果
- ② Cは桁上がりで、論理積の結果

入力		出力	
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



全加算器

① 全加算器は、

① 2桁以上の2進数の加算を行う場合において、
最下位の位以外の上位の各桁において、
下位からの桁上げによる入力を
考慮した加算回路である。

② 下位からの桁上がりと
その桁の2つの入力の3つの入力で、
桁内の加算と桁上がりの
2つの出力を得る回路となる。

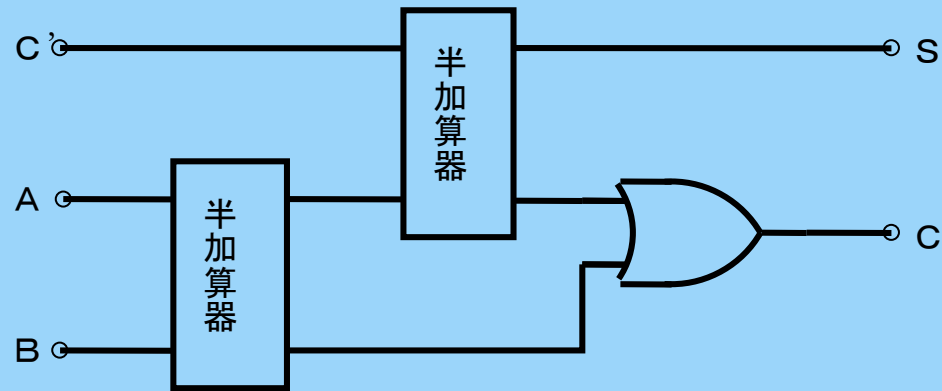
- ② 2個の半加算器とOR回路の
組合せからできている。
- ③ 真理値表に示す
入力値の組合せ8通りに対して、
4通りの出力値を得る。

全加算器出力の特徴

- ① 出力Sの値は
3入力のうち1のビットが奇数個の場合は1である。
- ② 出力Sの値は
3入力のうち1のビットが偶数個の場合は0となる。
- ③ 出力Cの値は
3入力のうち1のビットが2個以上あれば1である。
- ④ 出力Cの値は
3入力のうち1のビットが2個未満では0となる。

全加算器の真理値表と論理回路

入力			出力	
A	B	C	S	C
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1



デコーダ

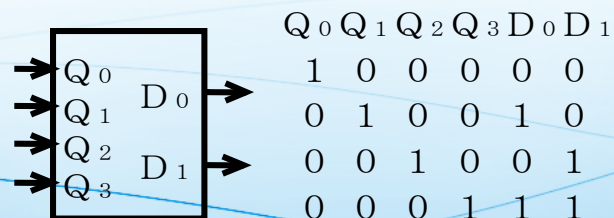
- ① デコーダは、
符号化された信号を解読する組み合わせ回路である。
- ② CPU命令の解読、
周辺回路の選択信号の作成などに使用する。
- ③ デコーダの出力は
相互に排他的であり、
入力パターンに対応する出力だけ1とし、
残りはすべて0となる。

D 0	Q 0
	Q 1
D 1	Q 2
	Q 3

D0	D1	Q0	Q1	Q2	Q3
0	0	1	0	0	0
1	0	0	1	0	0
0	1	0	0	1	0
1	1	0	0	0	1

エンコーダ

- ① エンコーダは、デコーダの反対の機能で、
符号化機能をもつ。
- ② 複数の入力のうち1つだけが1となり、
1になった入力に対応する出力パターンを生成する。



Q 0

D 0

Q 1

Q 2

Q 3

D 1

Q0	Q1	Q2	Q3	D0	D1
----	----	----	----	----	----

1	0	0	0	0	0
---	---	---	---	---	---

0	1	0	0	1	0
---	---	---	---	---	---

0	0	1	0	0	1
---	---	---	---	---	---

0	0	0	1	1	1
---	---	---	---	---	---

フリップフロップ回路

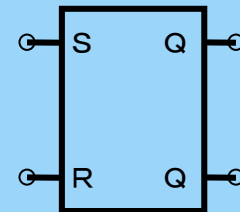
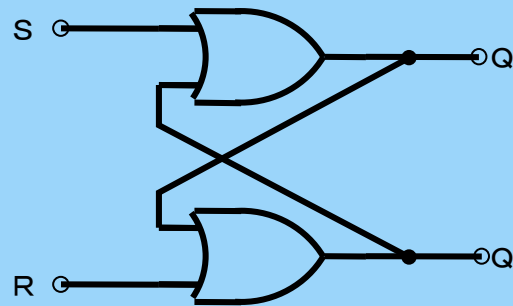
- ① フリップフロップ回路は、
0と1の二つの安定状態をもつ回路で、
順序回路の基本構成要素である。
- ② 2つの状態をとり、
どちらの状態にも安定する回路であり、
ある入力信号によって一方の状態に安定すると、
次に他方の状態に変える入力信号がくるまで、
その状態を保つ。

③ 1ビットのデータを記憶できる。

④ SRAMの記憶セルに使用する。

入力		出力	
S	R	Q	Q
0	0	—	—
1	0	1	0
0	1	0	1

— 変化なし



- * 指示を与えない限り同じ状態を保持する。
- * 2つの出力は常に反対を出力する。

オートマトン

① オートマトンは、

- ① 状態遷移を持つ順序理論で構成された
仮想的な機械である。
- ② 入出力装置と状態制御装置で構成される。

② オートマトンの4状態で構成

① 初期状態

② 受理状態

③ 入力

④ 状態変化

③ データを得る・計算する・結果を出力するなど、
コンピュータを使用して問題解決する場合の
処理手順をモデル化したものである。

④ 形式言語理論、文字列検索、
コンパイラの構文解析などの分野に応用される。

状態遷移図

- ① 状態遷移図は、
システムの取り得る状態の種別と
その状態が遷移するための要因との
関係をわかりやすく表現する記述形式である。
- ② 基本的な表記要素
 - ① 円
 - ② 矢印
 - ③ 遷移条件

状態遷移図の構成要素

① 状態

ある時点のプロセスの状態を円で表す。

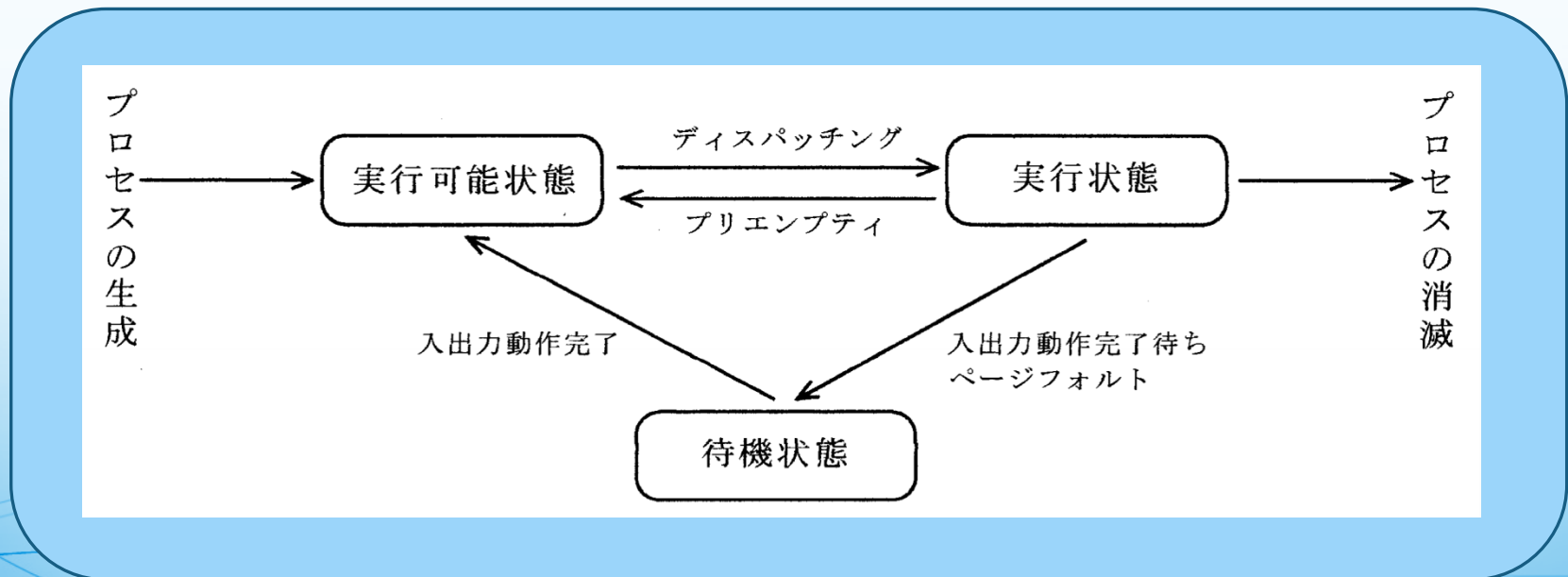
② 状態遷移

ある状態から他の状態への遷移を矢印で表す。

③ 遷移条件と処理

ある状態から他の状態へ遷移する条件と、
そのときに発生する処理を矢印の上に明記する。

タスク管理の状態遷移図



自動販売機の状態遷移図

- ① 自動販売機は、150円でジュースを販売している。
- ② 状態遷移1
 - ① 初期状態S0
 - ② 50円を投入すると状態S1に遷移
 - ③ 更に100円を投入すると、
ジュースを販売して初期状態に戻る。

③ 状態遷移2

- ① 初期状態で100円を投入
- ② 状態S2に遷移
- ③ 更に50円投入すると、
ジュースを販売して初期状態に戻る。

④ 状態遷移3

- ① 状態S2において、更に100円を投入する
- ② ジュースを販売すると共に、
お釣り50円を出力して、初期状態に戻る。

