

## IV 論理的なインタフェースの設計

### 学習目標

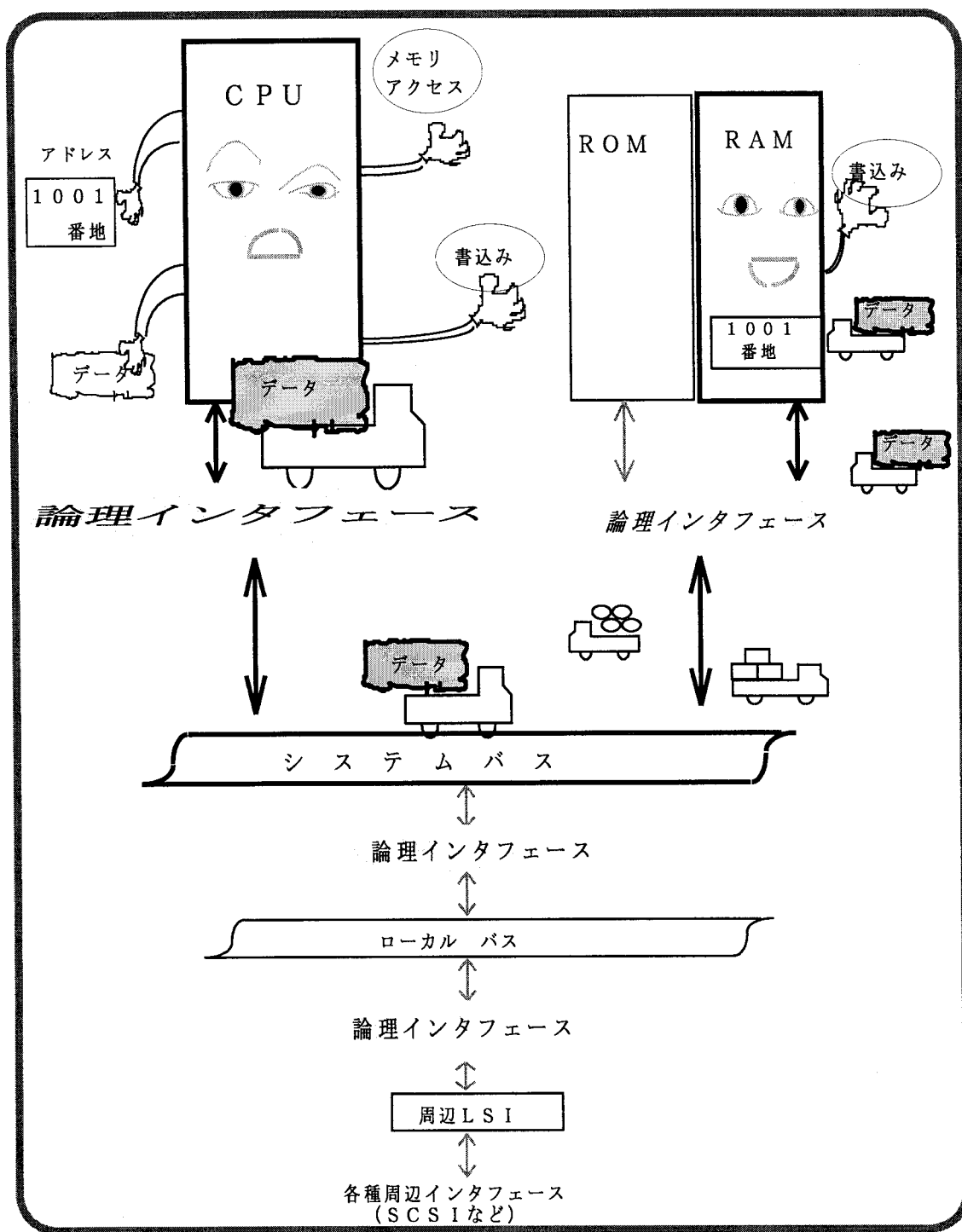
- 1 論理演算について真理値表を用いて理解させる。
- 2 正論理と負論理について理解させる。
- 3 簡単な組合わせ論理回路の解析と設計をさせる。
- 4 順序回路の動作を理解させる。
- 5 基本的なフリップフロップ回路を理解させる。
- 6 カウンタとレジスタについて理解させる。

### 学習の項目と内容

項 目	内 容
1 組合せ論理回路の基礎	組合せ論理回路とは 論理演算素子の種類 論理回路の動作
2 正論理と負論理	
3 組合わせ回路の活用	7セグメント表示データへの変換回路 アドレスデコード回路への応用
4 順序回路の基礎	順序回路とは 順序回路の動作 フリップフロップ
5 順序回路の例	RS_FF JK_FF D_FF T_FF
6 順序回路の活用	カウンタ レジスタ

# 1 組合せ論理回路の基礎

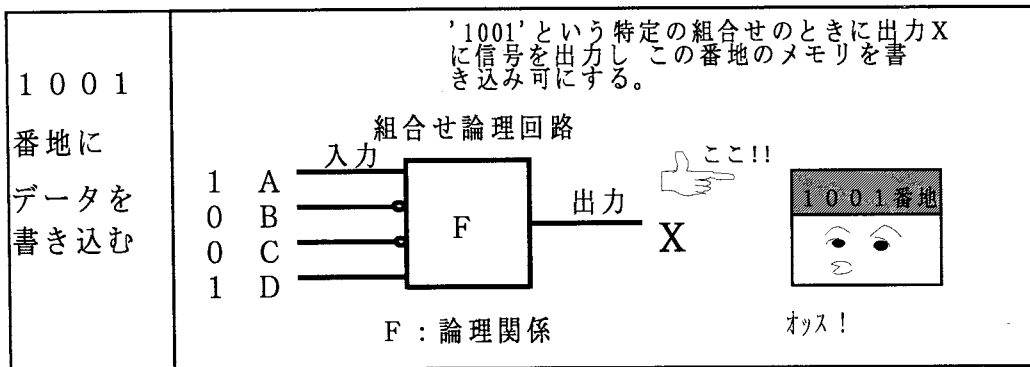
マイコン応用システムは、図IV-1のようなたくさんの論理的インタフェースによって構成されている。この論理的インタフェースは論理回路によって作られる。



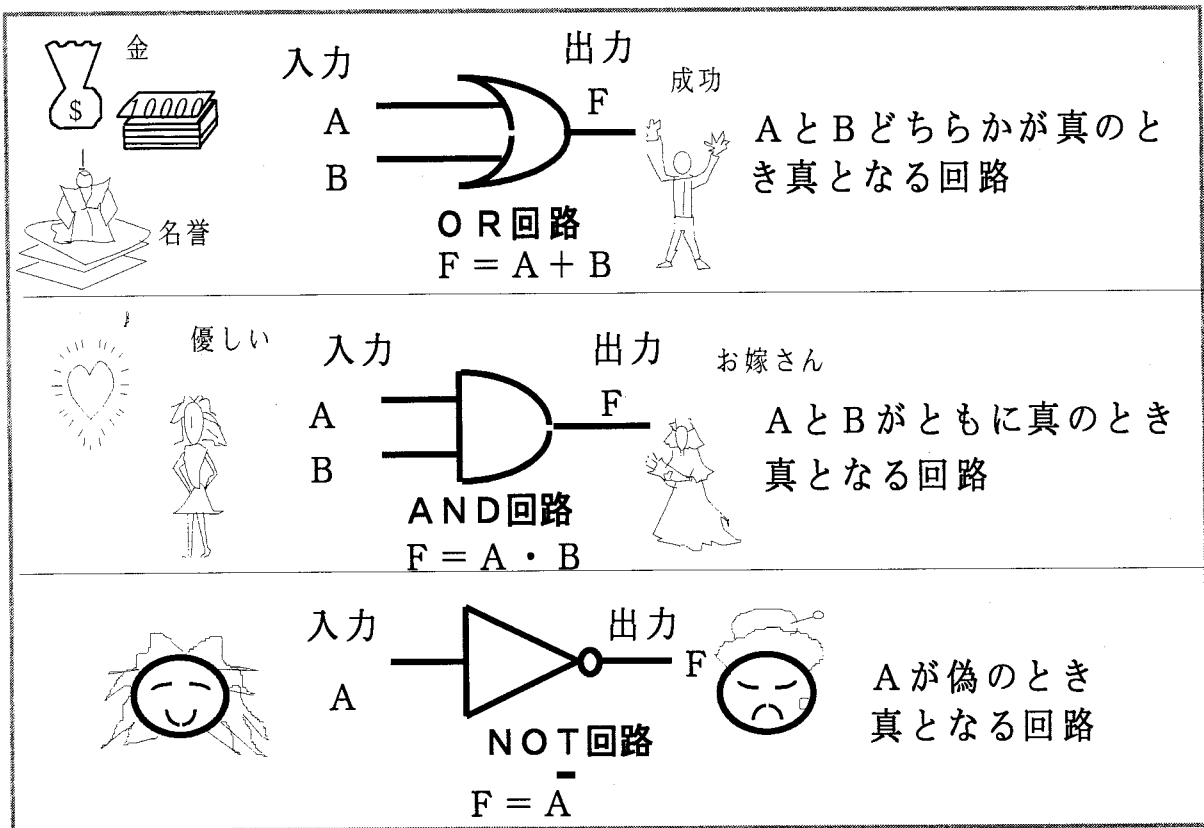
図IV-1 論理インタフェース

論理回路には、組合せ回路と順序回路の2種類がある。組合せ論理回路は、判定回路やアドレスデコード回路などのコード変換回路として使われる。順序回路は、メモリやI/Oへのデータの読み書きのタイミング回路やレジスタ、カウンタに使われる。

(1) 組合せ論理回路



図IV-2 組合せ回路



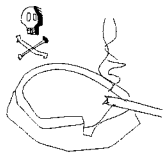
図IV-3 最も基本的な組合せ回路

組合せ論理回路とは、入力信号の組合せに対して常に定まった出力を出す論理回路を組合せ論理回路という(図IV-2)。また、入力信号が論理関係Fを満たしたときに真の値を出力する。このことを出力がアクティブになったという。加算回路・一致回路・乗算回路・デコーダ(特定の入力記号の組合せにより出力の一つを選択する回路)・符号器(入力信号に対して、特定のコードを生成する回路)などがあるが、いずれも基本的には、論理和・論理積・否定を実現する回路(図IV-3)の組合せで構成されている。

(2) その他の基本的な論理演算回路

論理関係を表す論理素子には、前述のAND回路・OR回路・NOT回路のほかに、実用面からよく使われるNAND回路・NOR回路・エクスクルーシブOR回路(EOR回路)がある(図IV-4)。

論理名称	論理関数	真理値表			M I L 記号
		入力		出力	
		A	B	X	
AND回路	$F = A \cdot B$	0	0	0	
		0	1	0	
		1	0	0	
		1	1	1	
OR回路	$F = A + B$	0	0	0	
		0	1	1	
		1	0	1	
		1	1	1	
NOT回路	$F = \overline{A}$	0	—	1	
		1	—	0	または  や
論理積の否定 NAND回路	$F = \overline{A \cdot B}$	0	0	1	
		0	1	1	
		1	0	1	
		1	1	0	
論理和の否定 NOR回路	$F = \overline{A + B}$	0	0	1	
		0	1	0	
		1	0	0	
		1	1	0	
排他的論理和 EOR回路	$F = \overline{A \cdot B} + \overline{A \cdot B}$	0	0	0	
		0	1	1	
		1	0	1	
		1	1	0	

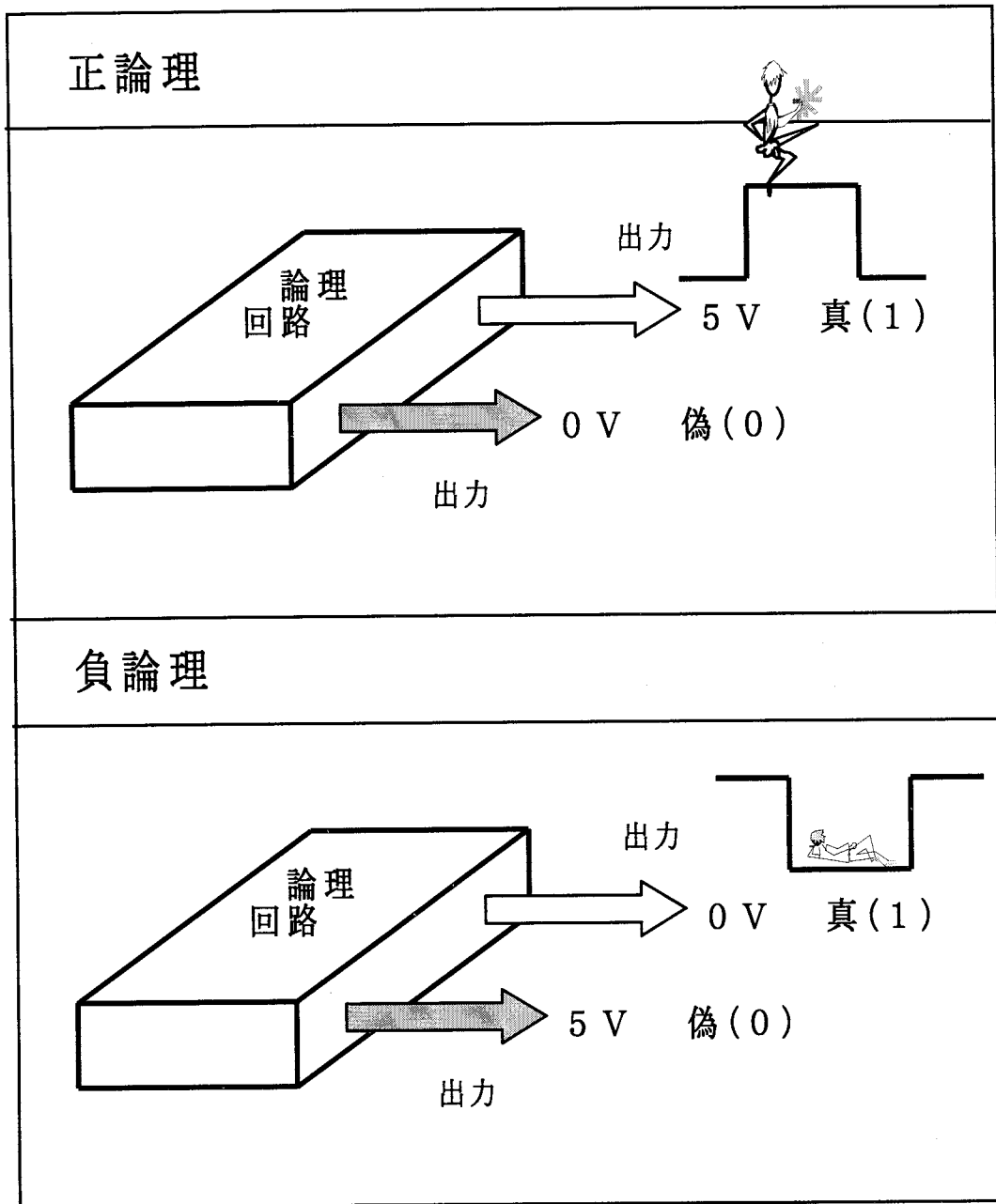


EORは、どちらか一つ...  
タバコか健康かどちらか一つ



図IV-4 その他の基本的な回路

## 2 正論理と負論理



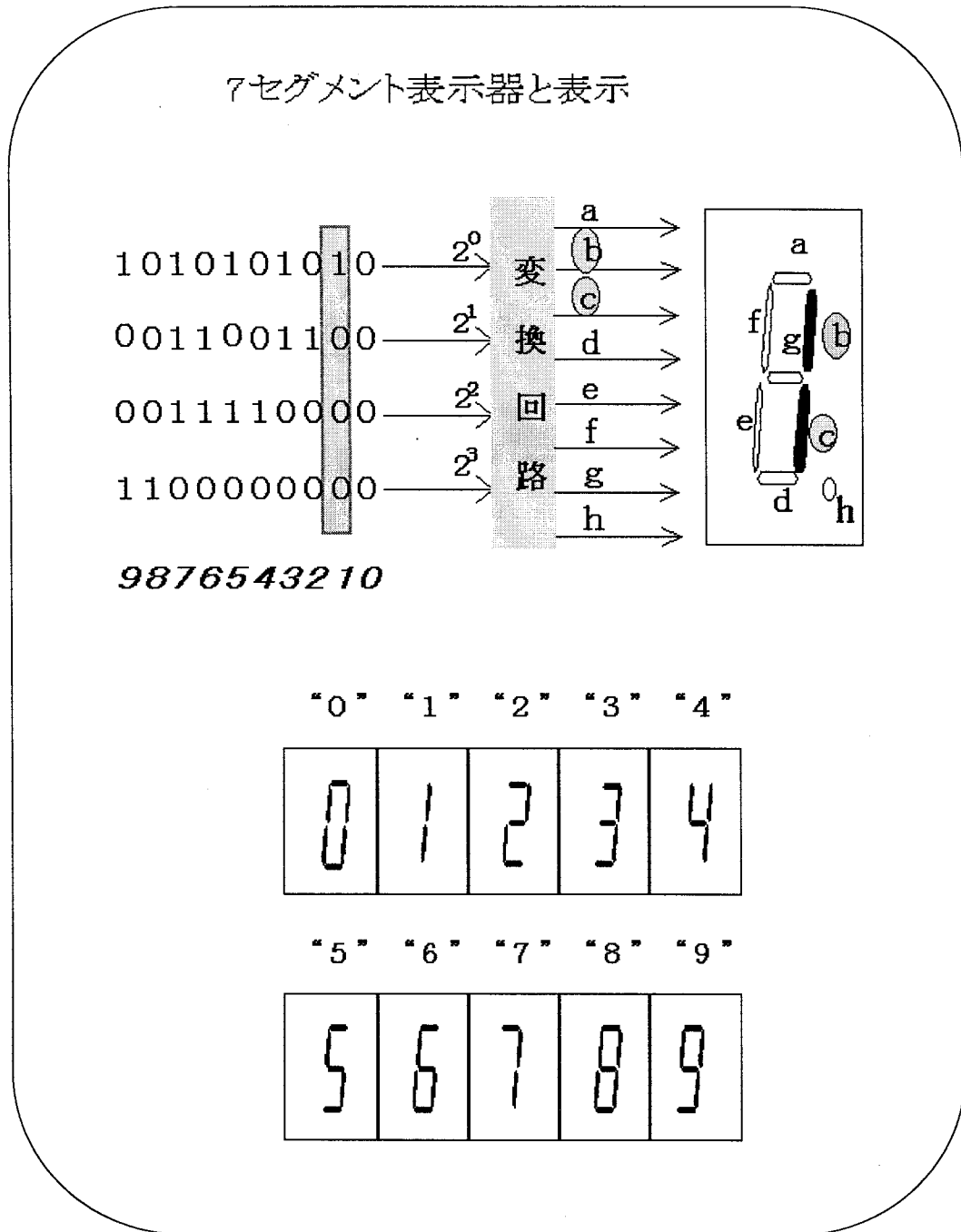
図IV-5 正論理と負論理

論理回路の0Vを偽(0)とみなす論理を正論理、5Vを偽(0)とみなす論理を負論理という。これは、歴史的にマイナスの電圧が論理回路に使われていたことがあり、その場合0を正(1)とみなしていたのである。実際の回路上では、GNDと同じ0Vが安定していて雑音に強いため、制御信号には負論理が使われていることが多い。また、特に断らないときは、5Vを1、0Vを0として扱っている。

### 3 組合せ論理回路の活用

#### (1) 7セグメント表示データへの変換回路

組合せ論理回路の応用例として、2進データを入力として受取り、これを図のような7セグメントの表示器に1桁の10進数字として表示させるための変換回路を設計する。



図IV-6 7セグメントLEDドライブ回路

表示要素は a から h まで 8 要素あるが、一つ一つの要素が点灯されるときの 2 進数の値を調べる。今、表示要素 'e' について調べてみると、'e' が点灯するのは、'0'、'2'、'6'、'8' である。2 進数の '0000'、'0010'、'0110'、'1000' に対応する。この 2 進数をゲート回路を使って表してみる。'0' に NOT の意味の  $\bar{\phantom{x}}$  を書き 4 本の入力を論理積の入力に入れ、その全体の出力を論理和で結んだ形になる。図 IV-7 (a) またこの回路を 'e' 以外の 'a'、'b'、'c'、'd'、'f'、'g'、'h' についても用意し、一つの IC に集積したものが変換回路全体になる (図 IV-7 (b))。

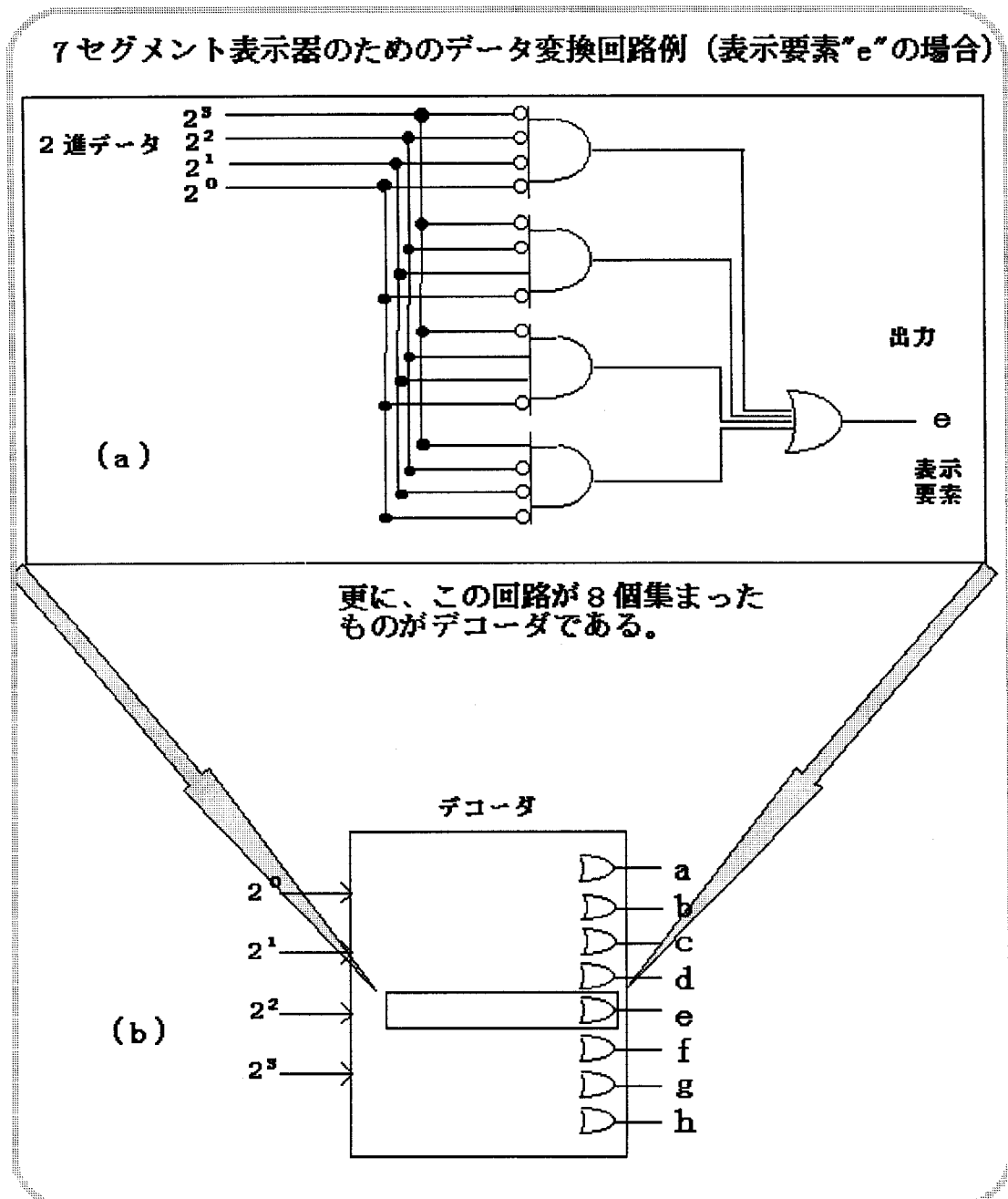
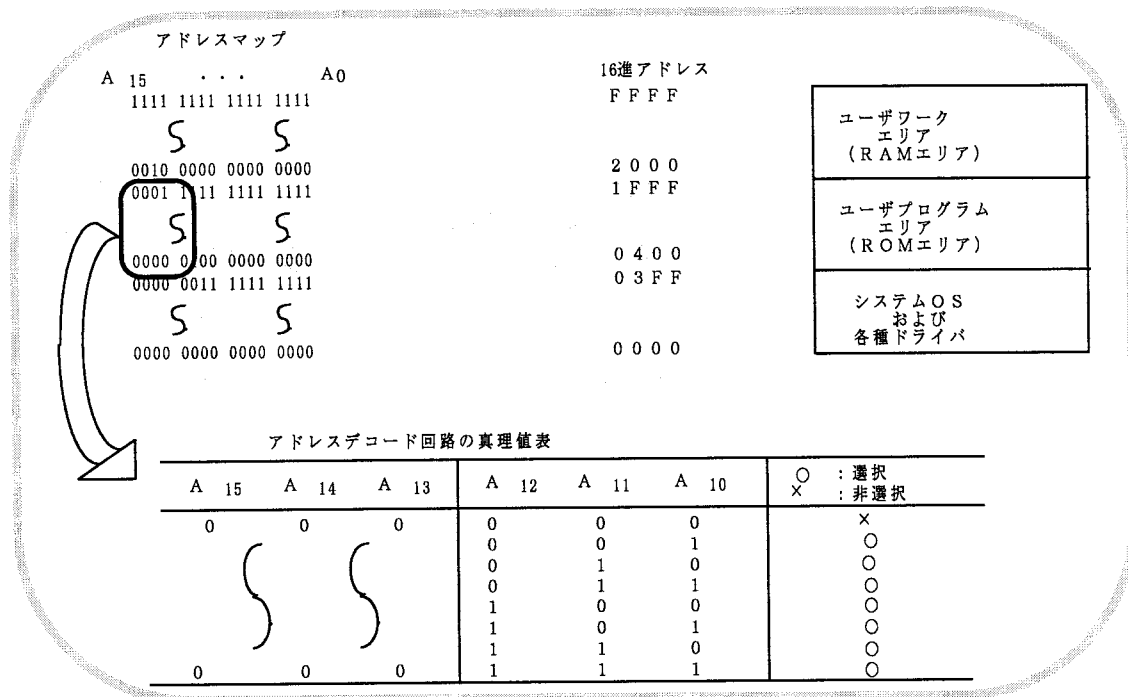


図 IV-7 変換回路

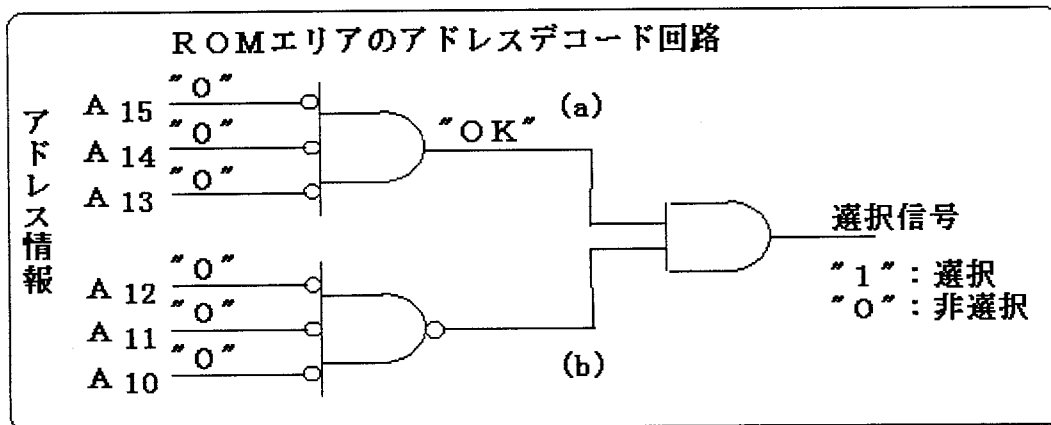
(2) アドレスデコード回路への応用

組合せ回路の代表的な例として、アドレスデコード回路があげられる。アドレスデコード回路は、アドレスを解読してメモリや周辺デバイスを選択する回路である。

アドレスマップのROMエリアを選択する回路を考える。



図IV-8 アドレスマップ



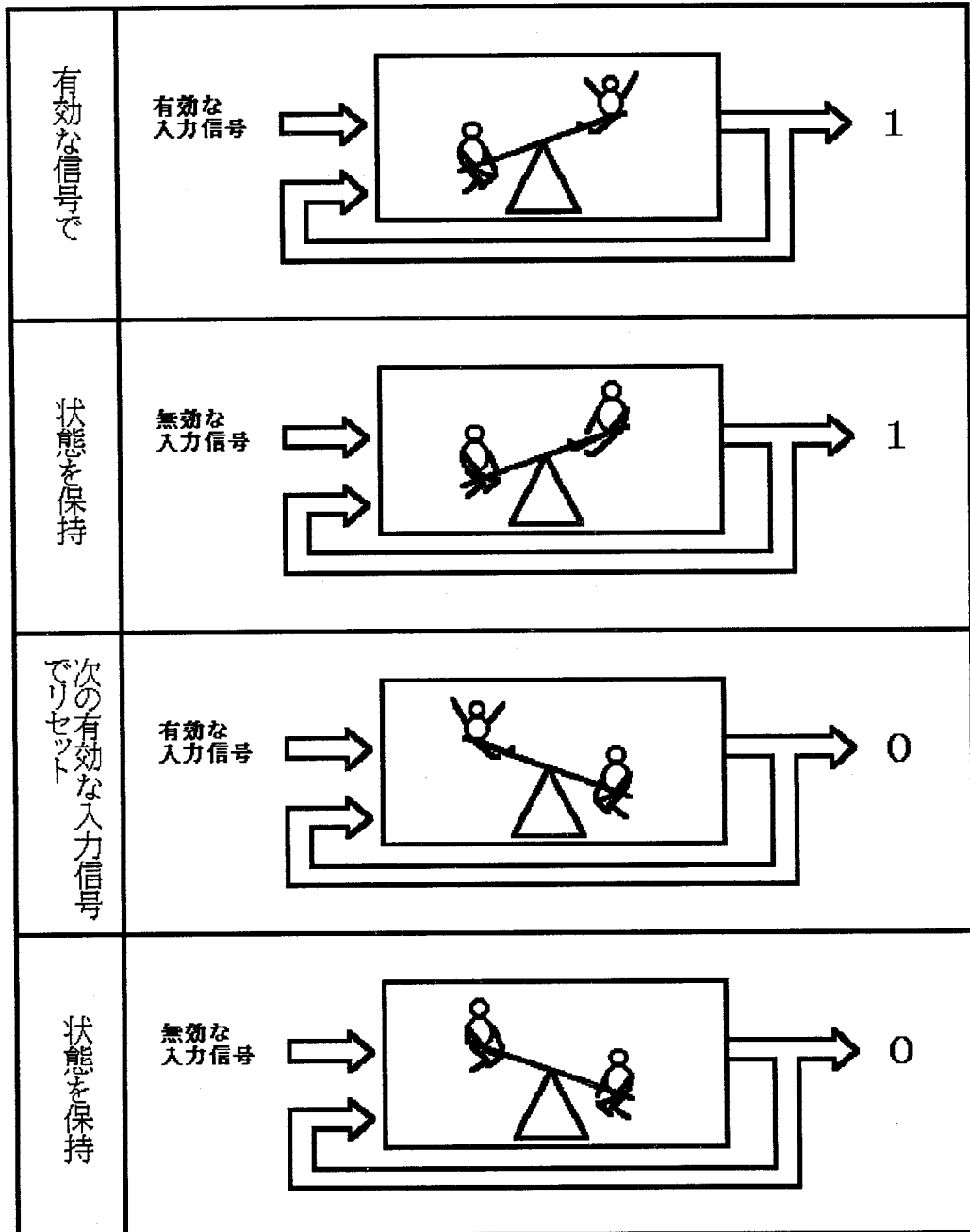
図IV-9 アドレスデコーダ

アドレスのどのビットからどのビットまでをデコード対象にすればよいかを考える。アドレスA0からA9は、様々な値をとるから、アドレスデコード回路の条件から除いて考える。アドレスA13からA15は常に"0"となるので、この部分を3入力の論理回路として構成する。残りのA10からA12をもう一つの論理回路として考える。A10からA12は、どれか一つが1のときにROMが選択されるので、A10=0、A11=0、A12=0のときだけ非選択とする。そして、最後にそれぞれの出力の論理積をとれば出来上がりである。



#### 4 順序回路の動作

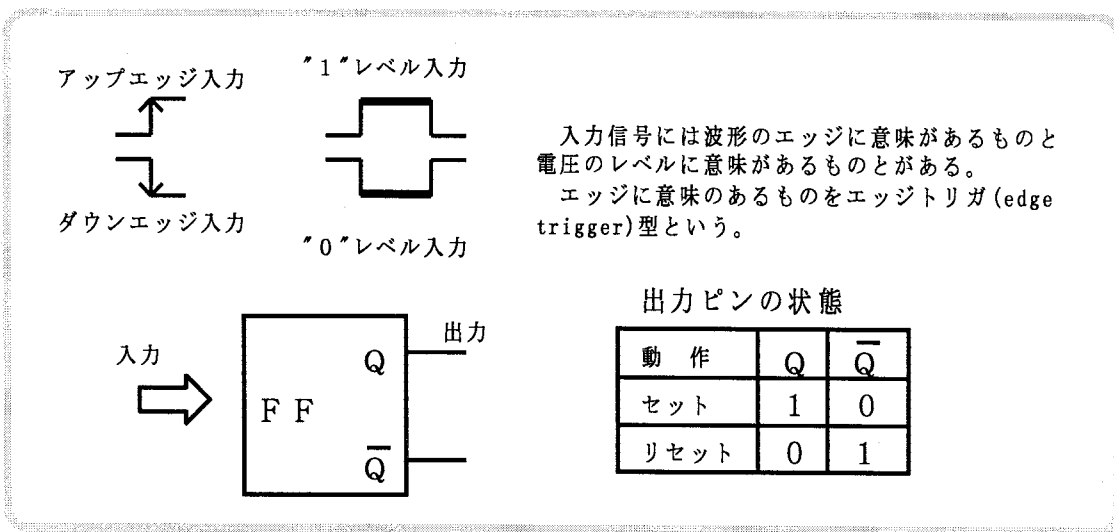
組合せ回路との違いは、時間的な要素を含んでいることである。順序回路は内部的状態を保持する機能を持ち、その内部状態と次の入力状態との組合せにより次に出力が決まる。また、次の有効な入力信号があるまでその状態を保持する。



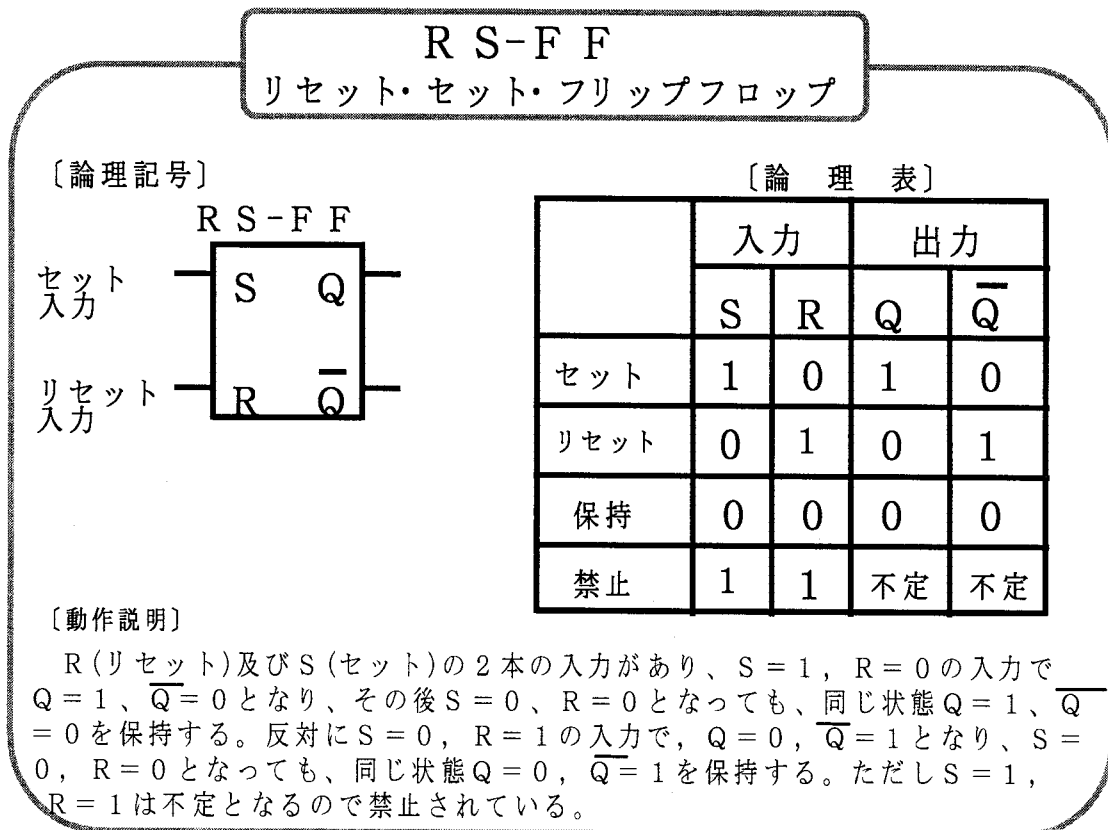
図IV-10 フリップフロップ回路の基本動作

この変化の状態を保持する機能を使い、一時記憶(レジスタ)やカウンタ等の回路に使われている。

## 5 順序回路の例(フリップフロップ)



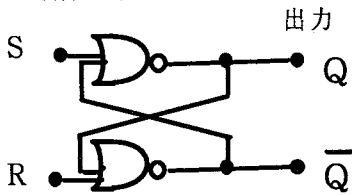
図IV-11 フリップフロップ回路



図IV-12 RS-FF

**NOR**  
ゲートで構成したRS-FF

〔論理記号〕



〔論理表〕

	入力		出力	
	S	R	Q	$\bar{Q}$
セット	1	0	1	0
リセット	0	1	0	1
保持	0	0	0	0
禁止	1	1	不定	不定

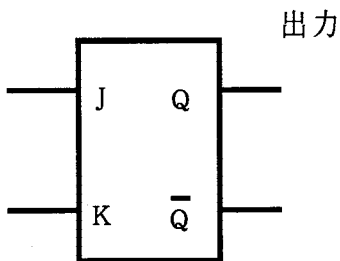
〔動作説明〕

NORで構成されたRS-FFは、セットが1でリセットが0のときである。また、 $S=R=1$ は不定となり動作が保証されないので禁止となる。

図IV-13 NORで構成したRS-FF  $Q_{n+1}$

**JK-FF**  
RS-FFの不定状態をなくし、反転機能を加えた。

〔論理記号〕



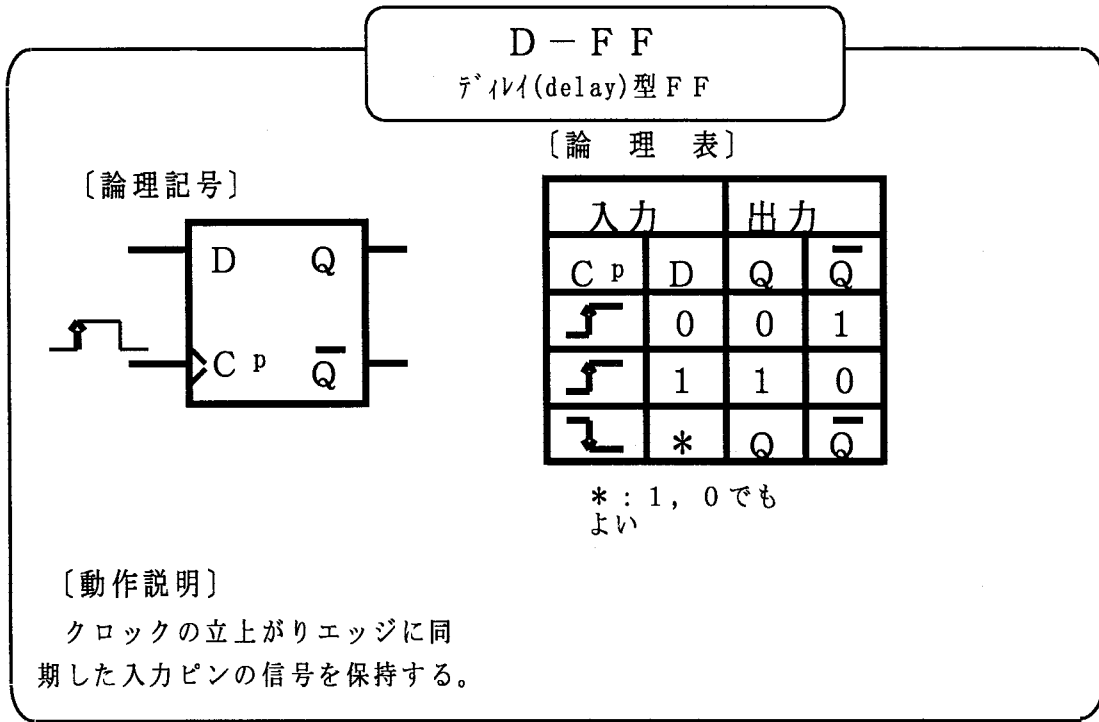
〔論理表〕

	入力		出力	
	J	K	$Q_{n+1}$	$\bar{Q}_{n+1}$
セット	1	0	1	0
リセット	0	1	0	1
保持	0	0	$Q_n$	$\bar{Q}_n$
反転	1	1	$\bar{Q}_n$	$Q_n$

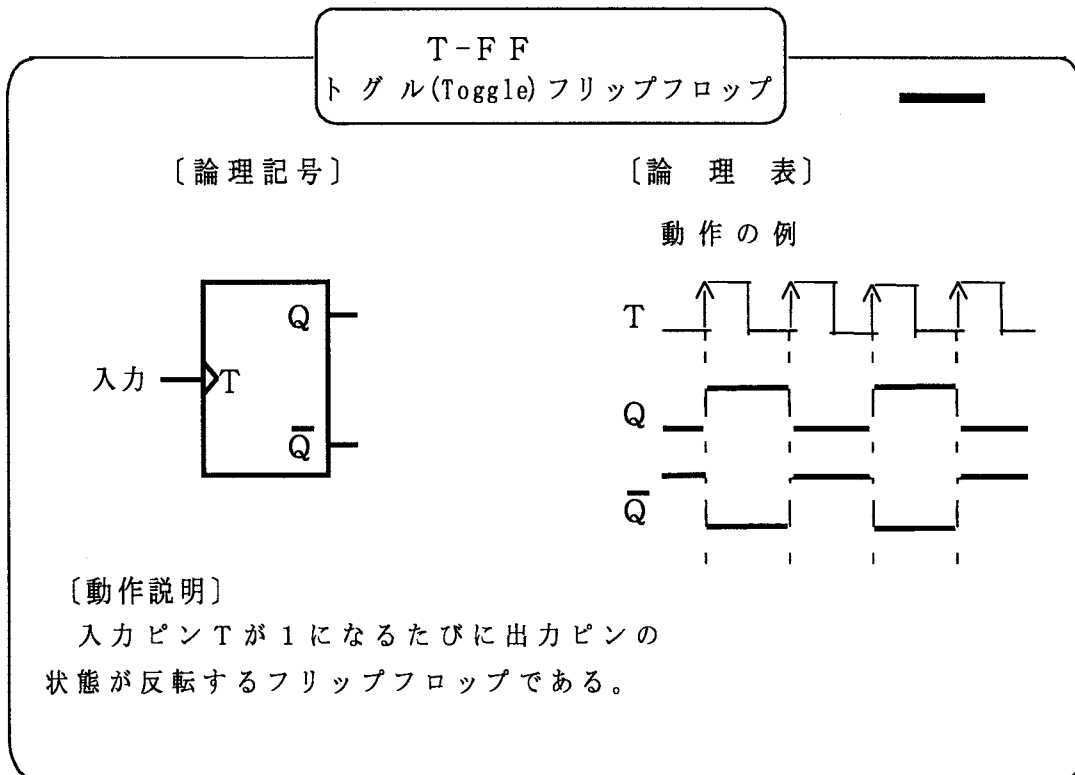
〔動作説明〕

二つの入力JとKがあり表のような特性がある。RS-FFの場合とは異なり $S=R=1$ のとき、入力が禁止でなく前の状態の反転となる。つまり、 $Q_n=1$ のときは、 $Q_{n+1}=0$ となる。

図IV-14 JK-FF



図IV-15 D-FF



図IV-16 T-FF

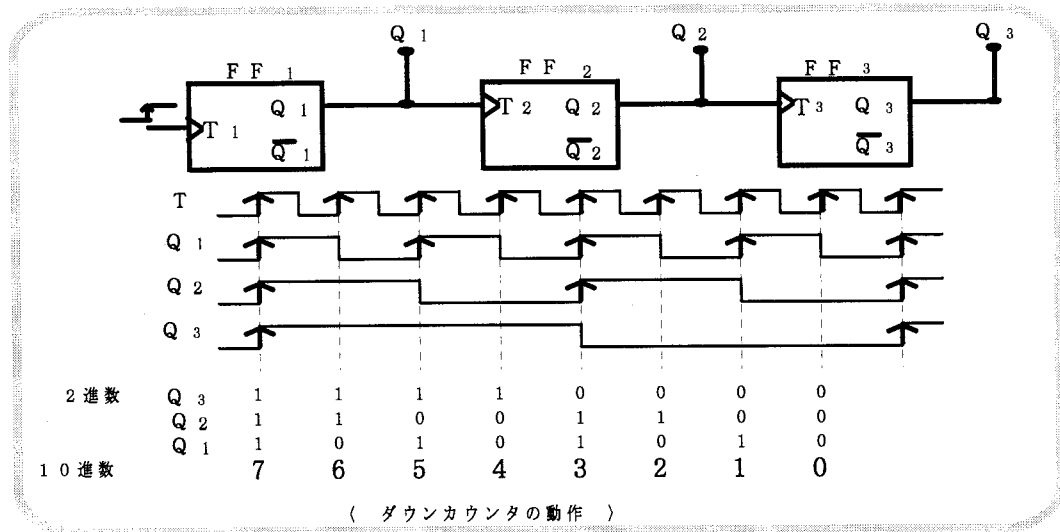
## 6 順序回路の活用

フリップフロップ回路を用いた順序回路の例としてカウンタとレジスタについて説明する。

### (1) カウンタ

#### ① n進カウンタ

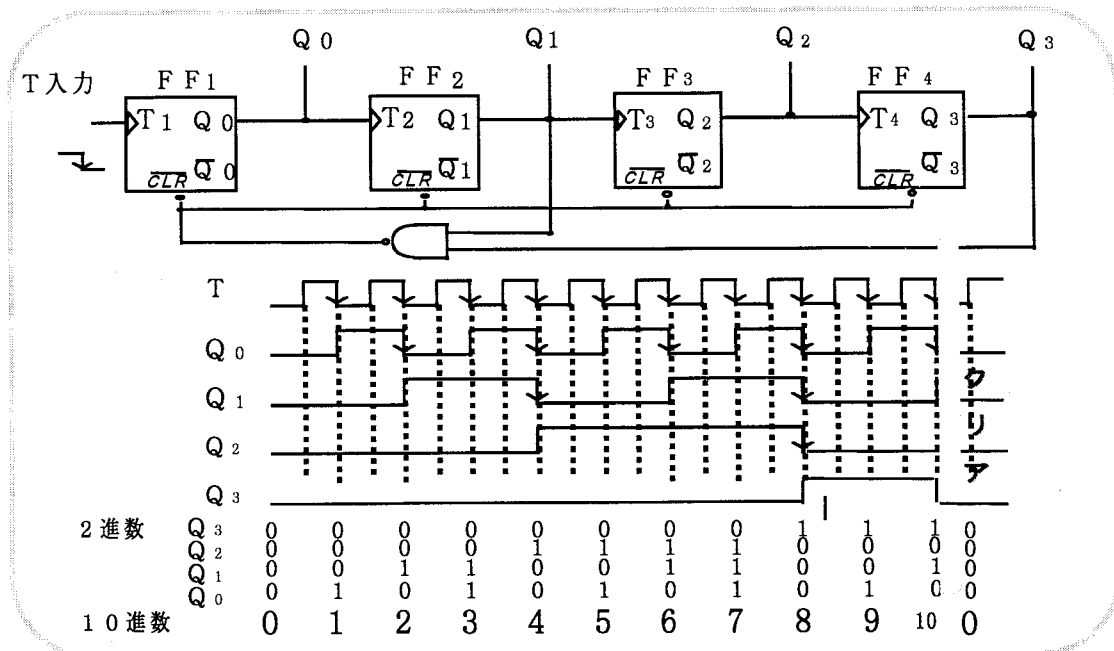
T-FFは、入力パルスの1/2の出力パルスを得られるので、T-FFをカスケード(直列)に接続するとn進のカウンタになる。



図IV-17 n進カウンタの基本

#### ② 10進カウンタ

2進カウンタに工夫を加え、Q<sub>0</sub>~Q<sub>3</sub>がある組合せのとき(1010)、T-FFをリセットすることで10進カウンタを表現している。

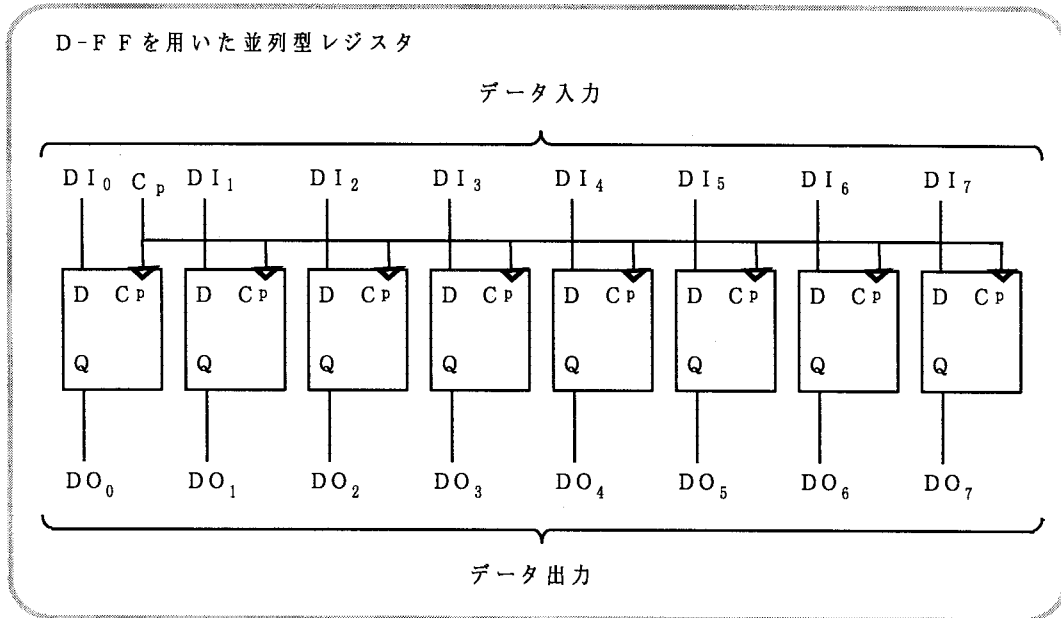


図IV-18 10進カウンタ

(2) レジスタ

① 8ビットレジスタ

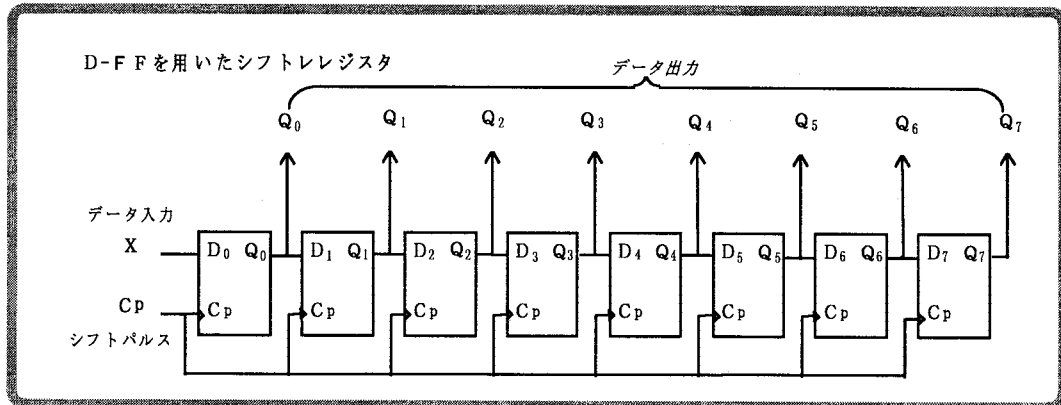
少量のデータを一時的に記憶する回路をレジスタ (register) という。ほとんどのフリップフロップは、保持状態を持っているので、一時記憶つまりレジスタとしての機能を持っている。その中でもレジスタとしてはD-F FやRS-F Fが使われる。



図IV-19 8ビットレジスタ

フリップフロップ回路一つでデータを1ビット記憶することができるから、これを必要とするビット数、つまりこの場合は8ビット分だけ並べれば、レジスタ回路を構成できる。

② シフトレジスタ



図IV-20 シフトレジスタ

クロックパルスが加えられるたびに入力信号Xは、右へ右へと順次シフトされる。今8ビットの直列信号を順次入力すると、入力が完了した時点で出力Q<sub>0</sub>Q<sub>1</sub>…Q<sub>7</sub>から8ビットの並列信号が得られる。