

# 論理回路（１）

---

---

# 基本論理演算と基本論理素子

---

# ブール代数と基本論理演算

---

- ▶ ブール代数(論理代数)
  - ▶ 「0」と「1」とで表す論理値に対する演算である論理演算を規定とする数学体系を、ブール代数あるいは論理代数という.
- ▶ 基本論理演算
  - ▶ 以下に示す3種類の論理演算を基本論理演算という.
    - ▶ 論理和 (OR, オア)
    - ▶ 論理積 (AND, アンド)
    - ▶ 否定 (NOT, ノット)

# 論理和

---

- ▶ 論理和 (OR)
  - ▶ 演算記号「+」を使う2項論理演算である.
  - ▶ 2個の演算項がいずれも 0 の場合にだけ, 演算結果が 0 である.
  - ▶ その他の演算項の組み合わせでは, 演算結果は 1 である.
- ▶  $0 + 0 = 0$
- ▶  $0 + 1 = 1$
- ▶  $1 + 0 = 1$
- ▶  $1 + 1 = 1$

# 論理積

---

- ▶ 論理積 (AND)
  - ▶ 演算記号「 $\cdot$ 」を使う2項論理演算である.
  - ▶ 2個の演算項がいずれも 1 の場合にだけ, 演算結果が 1 である.
  - ▶ その他の演算項の組み合わせでは, 演算結果は 0 である.
- ▶  $0 \cdot 0 = 0$
- ▶  $0 \cdot 1 = 0$
- ▶  $1 \cdot 0 = 0$
- ▶  $1 \cdot 1 = 1$

# 否定

---

- ▶ 否定 (NOT)
  - ▶ 演算記号「 $\overline{\phantom{x}}$ 」を使う単項論理演算である.
  - ▶ 1個の演算項の論理値を反転する.
- ▶  $\overline{0} = 1$
- ▶  $\overline{1} = 0$

# 基本論理素子

---

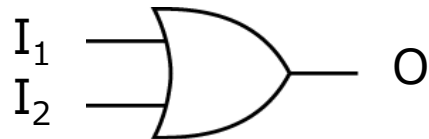
## ▶ 基本論理素子

- ▶ 3種類の基本論理演算の各機能を実現する最小単位のハードウェア機構を基本論理素子という.

- |               |                 |
|---------------|-----------------|
| ▶ 基本論理演算の演算項  | → 基本論理素子への入力信号  |
| ▶ 基本論理演算の演算結果 | → 基本論理素子からの出力信号 |

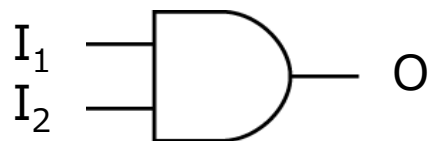
# 基本論理素子の記号表現と真理値表

## ■ OR素子



入力値 $I_1$	入力値 $I_2$	出力値 $O$
0	0	0
0	1	1
1	0	1
1	1	1

## ■ AND素子



入力値 $I_1$	入力値 $I_2$	出力値 $O$
0	0	0
0	1	0
1	0	0
1	1	1

## ■ NOT素子(インバータ)

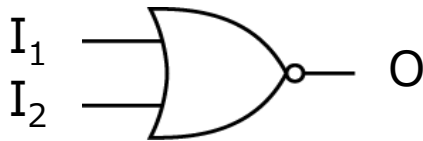


入力値 $I$	出力値 $O$
0	1
1	0



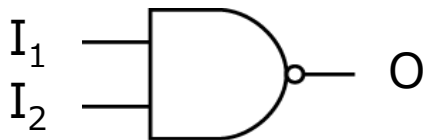
# 実用的論理素子の記号表現と真理値表（１）

## ■ NOR素子 (Not OR)



入力値 $I_1$	入力値 $I_2$	出力値 $O$
0	0	1
0	1	0
1	0	0
1	1	0

## ■ NAND素子 (Not AND)

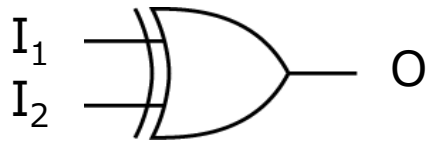


入力値 $I_1$	入力値 $I_2$	出力値 $O$
0	0	1
0	1	1
1	0	1
1	1	0

NOR素子・NAND素子の出力に付いている小さな丸印は、NOT素子（インバータ）を簡略化した記号である。

## 実用的論理素子の記号表現と真理値表（2）

### ■ XOR素子 (eXclusive OR)



入力値 $I_1$	入力値 $I_2$	出力値 $O$
0	0	0
0	1	1
1	0	1
1	1	0

### ■ XNOR素子 (eXclusive NOR)



入力値 $I_1$	入力値 $I_2$	出力値 $O$
0	0	1
0	1	0
1	0	0
1	1	1

# 万能論理素子

---

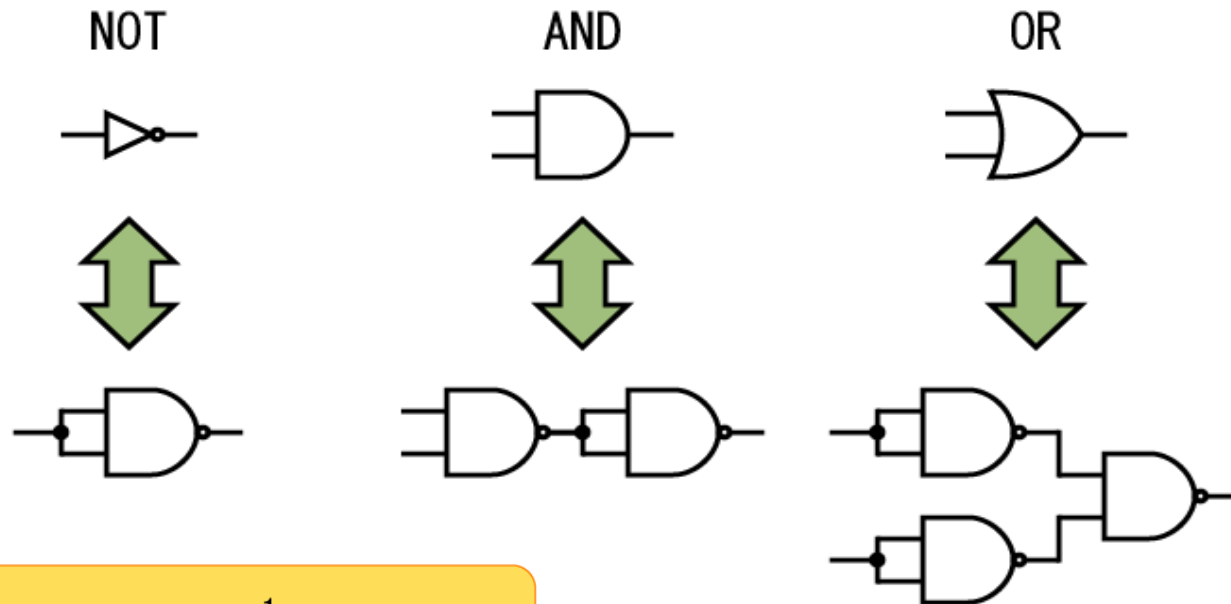
## ▶ 万能論理素子

- ▶ NAND素子とNOR素子は、それぞれ1種類だけで、基本論理素子(OR, AND, NOT)を実現できる.

あらゆる論理素子は、基本論理素子だけで構成することができる.  
すなわち、コンピュータハードウェアは、NAND素子、あるいは、  
NOR素子だけで、構成することができる.

# NAND素子による基本論理素子の実現

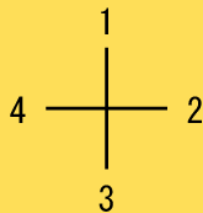
## ▶ NAND素子による基本論理素子の実現



補足事項



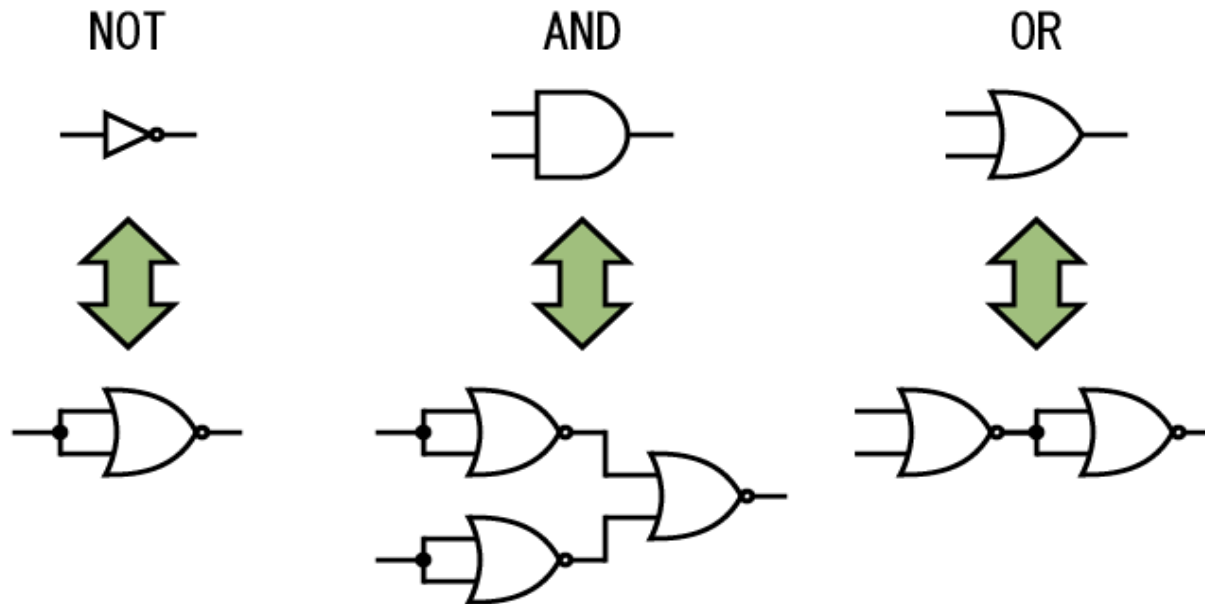
1, 2, 3, 4 は、  
すべて接続されている。



1, 3 は接続されている。  
2, 4 は接続されている。

# NOR素子による基本論理素子の実現

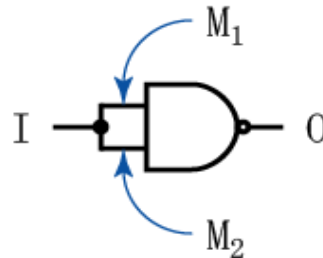
## ▶ NOR素子による基本論理素子の実現



## 演習問題

### ▶ 問題1

- ▶ NAND素子を用いて実現したNOT素子を以下に示す.
- ▶ 以下に示す真理値表を完成させよ.

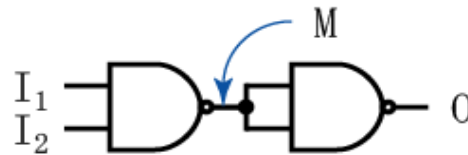


入力値 I	$M_1$	$M_2$	出力値 O
0			
1			

## 演習問題

### ▶ 問題2

- ▶ NAND素子を用いて実現したAND素子を以下に示す.
- ▶ 以下に示す真理値表を完成させよ.

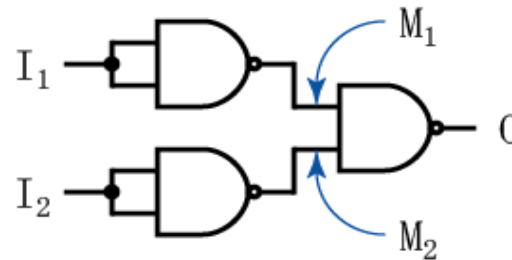


入力値 $I_1$	入力値 $I_2$	M	出力値 O
0	0		
0	1		
1	0		
1	1		

## 演習問題

### ▶ 問題3

- ▶ NAND素子を用いて実現したOR素子を以下に示す.
- ▶ 以下に示す真理値表を完成させよ.



入力値 $I_1$	入力値 $I_2$	$M_1$	$M_2$	出力値 $O$
0	0			
0	1			
1	0			
1	1			



# 論理回路

---

# 論理回路

---

## ▶ 論理回路

- ▶ 論理素子を信号線で接続(配線)して構成したある一定の働きをするハードウェア機構を論理回路という.
- ▶ コンピュータのハードウェア機構は, 論理素子を信号線で接続(配線)して構成される. すなわち, 論理回路である.

# 組み合わせ回路と順序回路

---

- ▶ 論理回路は、**組み合わせ回路**と**順序回路**の2つに分類することができる.
  - ▶ **組み合わせ回路**
    - ▶ 時刻  $t$  の出力が、時刻  $t$  の入力だけに依存するような論理回路.
    - ▶ なお、実際の論理素子では、**伝播遅延時間**(入力から出力へ信号が伝わるのに要する時間)を考慮する必要がある. すなわち、入力に応じて瞬時に出力が確定するわけではない.
  - ▶ **順序回路**
    - ▶ 時刻  $t$  の出力が、時刻  $t$  の入力だけでなく、過去の入力にも依存するような論理回路. (詳細は次週以降に学ぶ)

# 組み合わせ回路

---

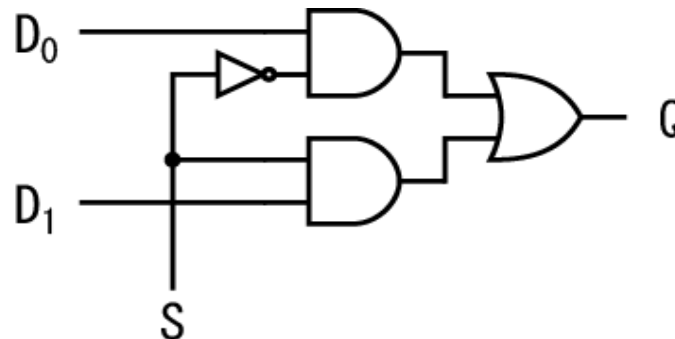
# マルチプレクサ

## ▶ マルチプレクサ

- ▶  $n$ 本の制御線によって,  $2^n$ 本の入力信号の中から1つの入力を選択して出力する機能を持つ組み合わせ回路.
- ▶  $2^n = m$ 本の入力信号の中から1つの入力を選択して出力するマルチプレクサを, 「 $m \times 1$ マルチプレクサ」という.
- ▶ なお, マルチプレクサのことを, データセレクトタともいう.

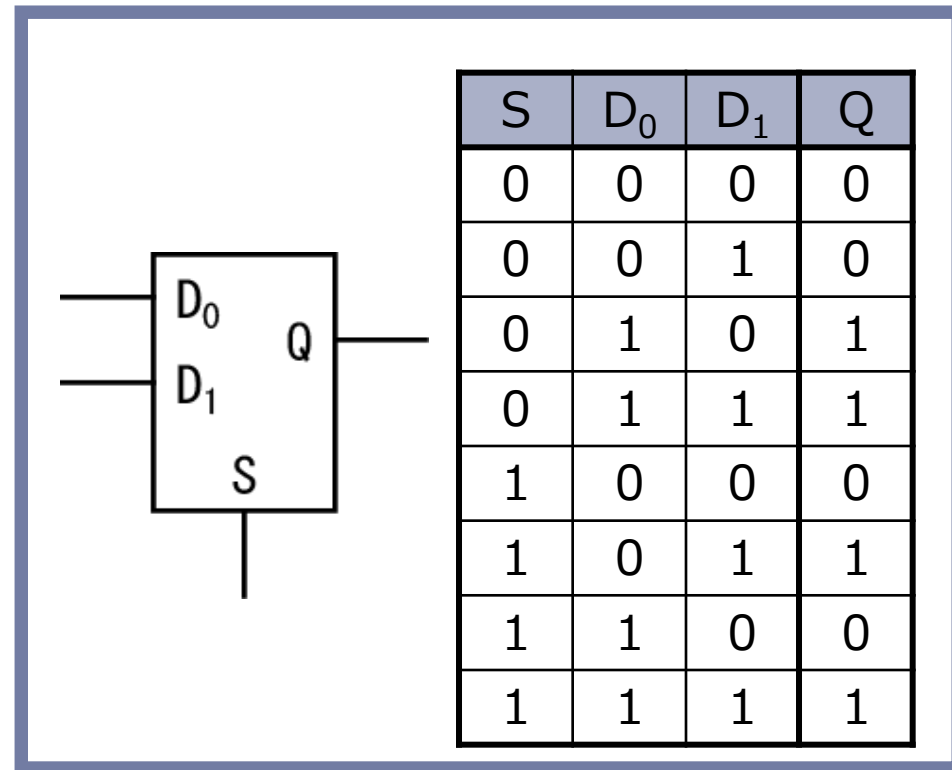
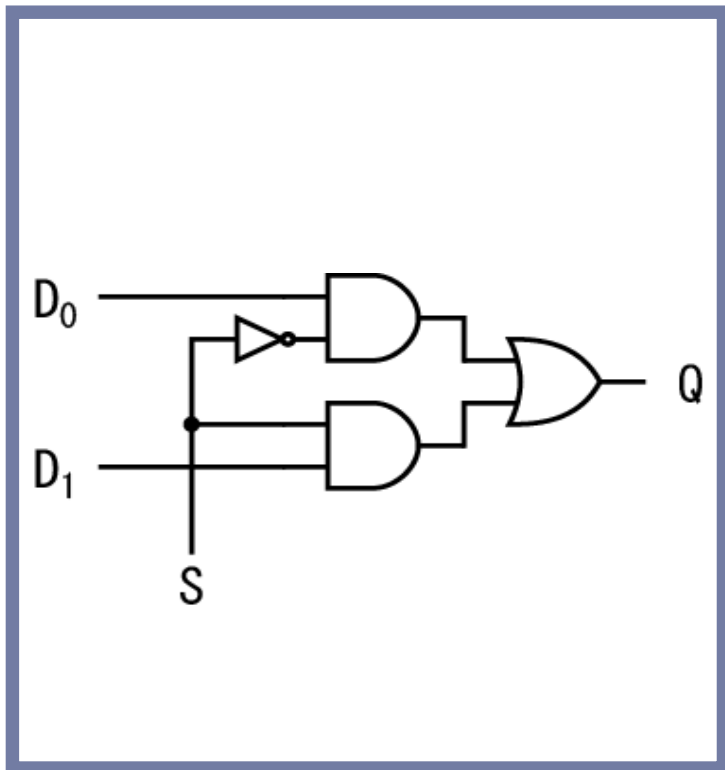
## ▶ 【例】 $n=1$ の場合

- ▶  $n=1$ の場合には,  $2^1=2$ となるので, 「 $2 \times 1$ マルチプレクサ」となる.
- ▶ 基本論理素子により構成した論理回路



## 組み合わせ回路の表現

- ▶ 一般に、組み合わせ論理回路は、以下のように、図と真理値表で表現される.



# 真理値表の単純化

- ▶ 真理値表において、値が0でも1でもかまわない場合、単純化を行う。
  - ▶ 入力のある組み合わせに対して、出力がどちらになってもかまわない場合には、出力の該当部分に「X」と表記する。
  - ▶ 入力の一部のみで出力が決まってしまう場合には、値がどちらでもかまわないような入力が存在する。その場合には、入力の該当部分に「X」と表記する。（以下の例を参照）

【例】 2×1マルチプレクサ

S	D <sub>0</sub>	D <sub>1</sub>	Q
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1



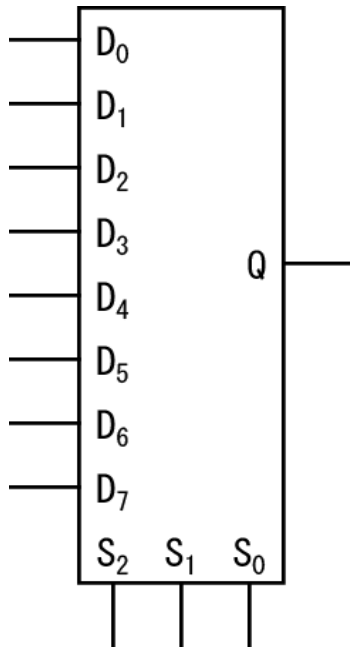
S	D <sub>0</sub>	D <sub>1</sub>	Q
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1



S	D <sub>0</sub>	D <sub>1</sub>	Q
0	0	X	0
0	1	X	1
1	X	0	0
1	X	1	1

# 8×1マルチプレクサ

## ▶ 8×1マルチプレクサ



S <sub>2</sub>	S <sub>1</sub>	S <sub>0</sub>	D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	Q
0	0	0	X	X	X	X	X	X	X	0	0
0	0	0	X	X	X	X	X	X	X	1	1
0	0	1	X	X	X	X	X	X	0	X	0
0	0	1	X	X	X	X	X	X	1	X	1
0	1	0	X	X	X	X	X	0	X	X	0
0	1	0	X	X	X	X	X	1	X	X	1
0	1	1	X	X	X	X	0	X	X	X	0
0	1	1	X	X	X	X	1	X	X	X	1
1	0	0	X	X	X	0	X	X	X	X	0
1	0	0	X	X	X	1	X	X	X	X	1
1	0	1	X	X	0	X	X	X	X	X	0
1	0	1	X	X	1	X	X	X	X	X	1
1	1	0	X	0	X	X	X	X	X	X	0
1	1	0	X	1	X	X	X	X	X	X	1
1	1	1	0	X	X	X	X	X	X	X	0
1	1	1	1	X	X	X	X	X	X	X	1

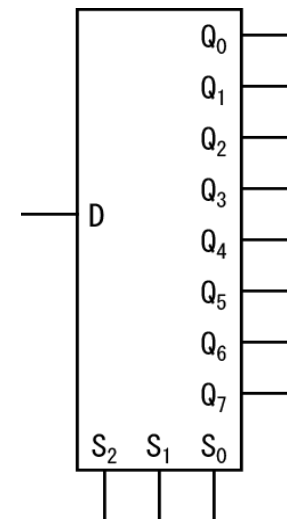


# デマルチプレクサ

## ▶ デマルチプレクサ

- ▶ マルチプレクサと反対の機能を持つ組み合わせ回路.
- ▶  $n$ 本の選択(制御)線によって,  $2^n$ 本の出力の中から1つの出力を選択して, その出力のみに, 入力信号を出力する.
- ▶ 1つの入力と,  $2^n = m$ 本の出力を持つデマルチプレクサを, 「 $1 \times m$ デマルチプレクサ」という.
- ▶ なお, デマルチプレクサのことを, データ分配器ともいう.

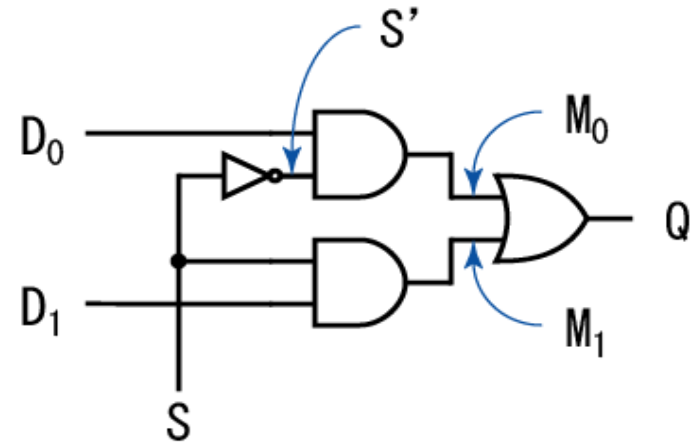
- ▶ 【例】  $1 \times 8$ デマルチプレクサ  
(真理値表は省略)



# 演習問題

## 問題4

- 右に示す2×1マルチプレクサの真理値表を完成させよ.



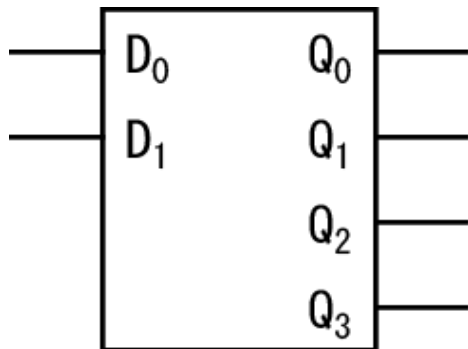
$D_0$	$D_1$	$S$	$S'$	$M_0$	$M_1$	$Q$
0	0	0				
0	1	0				
1	0	0				
1	1	0				
0	0	1				
0	1	1				
1	0	1				
1	1	1				

# デコーダ

## ▶ デコーダ

- ▶ コードの解読(復号化, デコード)を行う機能を持つ組み合わせ回路.
- ▶ 入力ビットパターンに対応する出力だけを1とし, 残りすべての出力を0とする.
- ▶  $n$ 入力と,  $2^n=m$ 出力を持つデコーダを, 「 $n \times m$ デコーダ」という.

## ▶ 【例】 $n=2$ の場合 $2 \times 4$ デコーダ



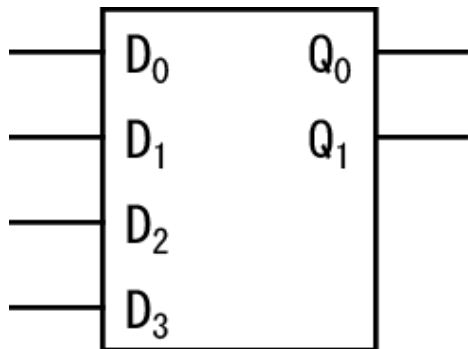
$D_1$	$D_0$	$Q_3$	$Q_2$	$Q_1$	$Q_0$
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

# エンコーダ

## ▶ エンコーダ

- ▶ デコーダの反対の機能(符号化, エンコード)を持つ組み合わせ回路.
- ▶ 複数入力の中の1つの入力だけが1となり, 1になった入力に対応する出力ビットパターンを生成する.
- ▶  $2^n = m$ 入力と,  $n$ 出力を持つエンコーダを, 「 $m \times n$ エンコーダ」という.

## ▶ 【例】 $n=2$ の場合 $4 \times 2$ エンコーダ



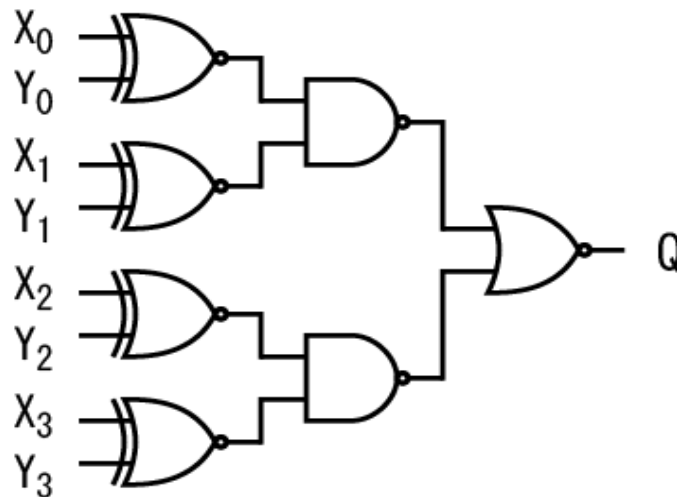
$D_3$	$D_2$	$D_1$	$D_0$	$Q_1$	$Q_0$
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

# コンパレータ

## ▶ コンパレータ

- ▶ XOR素子あるいはXNOR素子を複数個ならべて構成したデータ比較回路.
- ▶ 等しいか等しくないかを判定するコンパレータ(以下の例を参照), 等しいか等しくないかだけではなく大小比較の可能なコンパレータなどがある.

## ▶ 【例】 4ビットコンパレータ

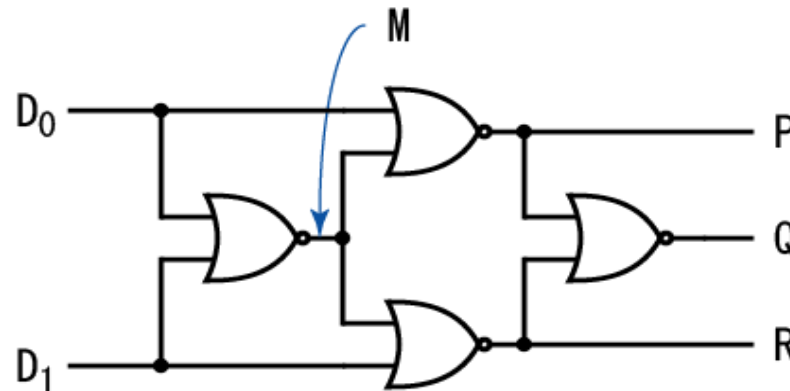


$$X \equiv Y \rightarrow Q=1$$

## 演習問題

### 問題5

- 右に示す1ビット大小比較器の真理値表を完成させよ



$D_0$	$D_1$	$M$	$P$	$Q$	$R$
0	0				
0	1				
1	0				
1	1				