

専門基礎科目 - 必修科目
GB10804 論理回路

1

システム情報系情報工学域 山口佳樹

専門基礎科目 - 必修科目（情報科学類）
基礎科目 - 関連科目（情報メディア創成学類）
基礎科目 - 関連科目（工学システム学類）

57

60

組み合わせ回路の設計法

- 論理和標準形(加法標準形)による設計
 - 最も簡単な手法
 - 論理関数的説明では、積項の和、に基づく設計
 - 無駄が多い
- カルノー図(Karnaugh Map)
 - 簡単化された回路を設計する手法で実用的
 - $H(1)$ の団体に対し1つのANDゲートを割り付ける
- (論理式による変形)

60

論理和標準形による設計

- 真理値表のH(1)レベルの行のチェック
- 入力信号とその反転記号を描く
- 入力信号がH(1) \rightarrow そのまま ANDゲートへ入力
- 入力信号がL(0) \rightarrow 反転記号からANDゲートへ入力
- ANDの出力をOR回路で結ぶ

真理値表から回路を論理和標準形で設計

- 論理積が項(最小演算単位)となる

C	B	A	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

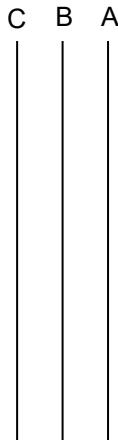


- 真理値表の1レベルの行のチェック

真理値表から回路を論理和標準形で設計

- 入力信号が 1 → そのまま ANDゲートへ入力

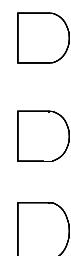
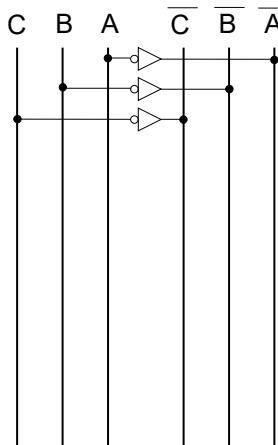
C	B	A	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0



真理値表から回路を論理和標準形で設計

- 入力信号が 1 → そのまま ANDゲートへ入力
- 入力信号が 0 → 反転記号から ANDゲートへ入力

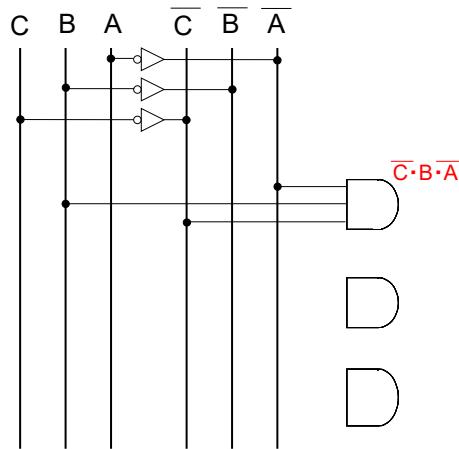
C	B	A	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0



真理値表から回路を論理和標準形で設計

- ・ 入力信号が 1 → そのまま ANDゲートへ入力
- ・ 入力信号が 0 → 反転記号から ANDゲートへ入力

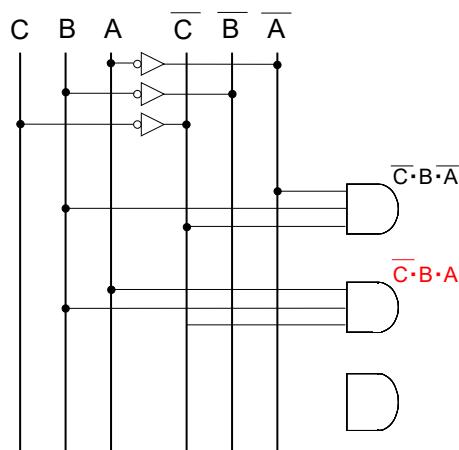
C	B	A	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0



真理値表から回路を論理和標準形で設計

- ・ 入力信号が 1 → そのまま ANDゲートへ入力
- ・ 入力信号が 0 → 反転記号から ANDゲートへ入力

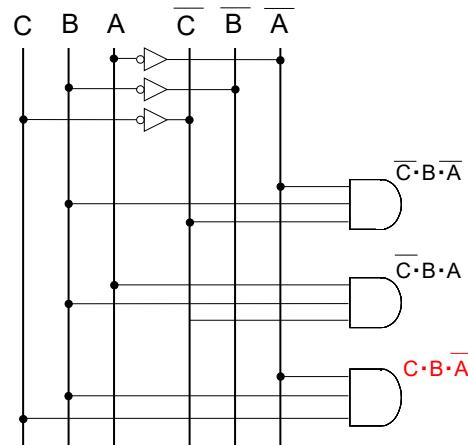
C	B	A	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0



真理値表から回路を論理和標準形で設計

- ・ 入力信号が 1 → そのまま ANDゲートへ入力
- ・ 入力信号が 0 → 反転記号から ANDゲートへ入力

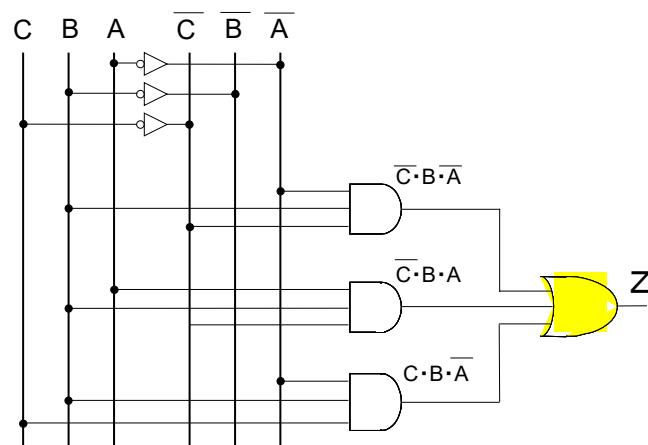
C	B	A	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0



真理値表から回路を論理和標準形で設計

- ・ ANDの出力をOR回路で結ぶ

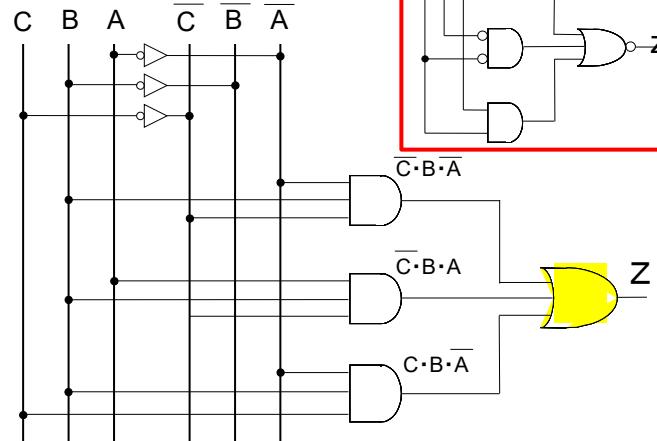
C	B	A	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0



真理値表から回路を論理和標準形で設計

- ANDの出力をOR回路で結ぶ

C	B	A	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0



演習問題①

三変数関数 f の真理値表が以下のとき、

① f を論理和標準形で示しなさい。

a	b	c	f
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

② f を論理積標準形で示しなさい。

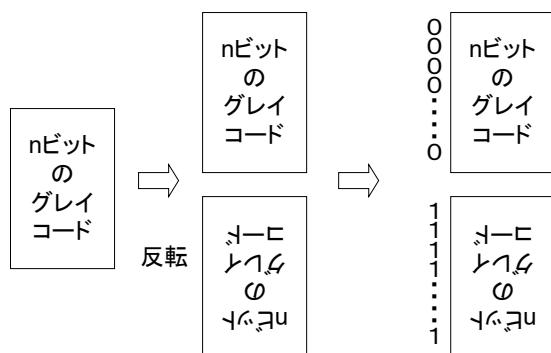
カルノー図による簡単化

- カルノー図の中の1が書かれているセルを出来るだけ少ない長方形(ループ)で囲む。
 - 囲んだ長方形の中のセルは、全て1(または- don't care)であること。
 - 囲んだ長方形の中のセルの数は、1, 2, 4, 8のように2のべき乗であること。
 - 同じセルを2つ以上のループで共有しても良い。
 - カルノー図の上下の端、および左右の端は連続していると考える。

グレイコード(Gray Code)

- となりの数と1ビットのみしか変化しない
 - カルノー図の長方形に囲む方式とマッチ
 - 専門的に言うと「ハミング距離が1」のコード
 - 元々、Reflected Binary Code と提案された。
(提案者の名前が Frank Gray だから Gray Code)
- グレイコードの作り方
 - (1) nビットの gray code を逆順に並び替えて、
元の n ビットに繋げて n+1 ビットにする。
 - (2) 2進コードからグレイコードへ変換する。

nビット → n+1 ビット のグレイコードの作成



2進コードからグレイコードへの変換

- 2進数を、1ビット右へ論理シフトした数を作成
- 元の数と排他的論理和(XOR)をとる

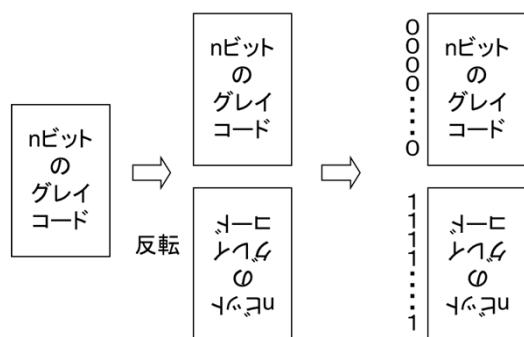
右へ論理シフト		
000	000	000
001	000	001
010	001	011
011	XOR 001	010
100	010	110
101	010	111
110	011	101
111	011	100

グレイコード(Gray Code)

- 4 bit のグレイコードを作成しなさい。

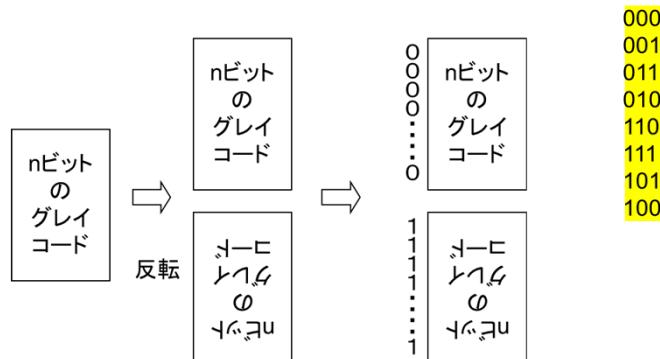
グレイコード(Gray Code)

- 4 bit のグレイコードを作成しなさい。



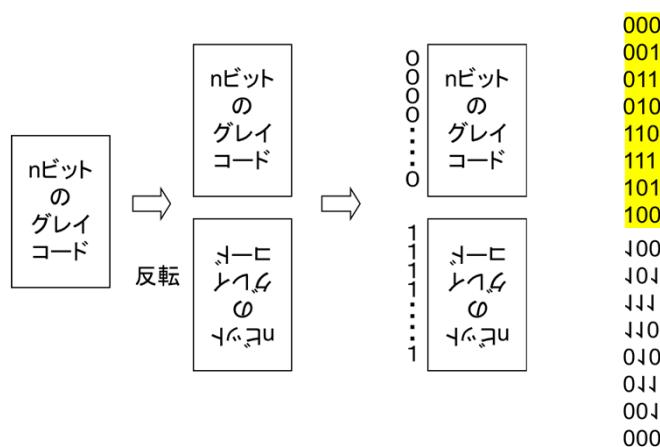
グレイコード(Gray Code)

- 4 bit のグレイコードを作成しなさい。



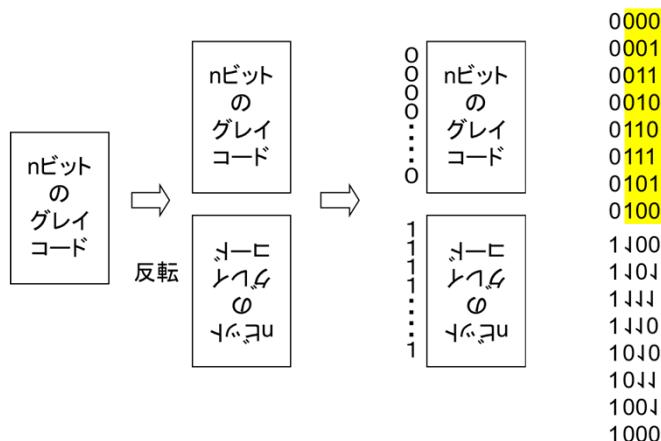
グレイコード(Gray Code)

- 4 bit のグレイコードを作成しなさい。



グレイコード(Gray Code)

- 4 bit のグレイコードを作成しなさい。



カルノー図

- 論理回路(論理式)を簡単化するための表
- グレーコードを利用して作成

次のカルノー図を簡単化しよう

BA DC	00	01	11	10
00	1		1	1
01		1		
11		1		
10			1	

次のカルノー図を簡単化しよう

BA DC	00	01	11	10
00	1		1	1
01		1		
11		1		
10			1	

次のカルノー図を簡単化しよう

BA DC	00	01	11	10
00	1		1	1
01		1		
11		1		
10			1	

次のカルノー図を簡単化しよう

BA DC	00	01	11	10
00	1		1	1
01		1		
11		1		
10			1	

次のカルノー図を簡単化しよう

	BA DC	00	01	11	10
00					
01		1	1	1	
11			1	1	
10				1	

次のカルノー図を簡単化しよう

	BA DC	00	01	11	10
00				1	
01		1			1
11		1	1	1	
10		1	1		

次のカルノー図を簡単化しよう

BA DC	00	01	11	10
00	1	1	1	1
01	1		1	1
11			1	
10	1	1	1	

次のカルノー図を簡単化しよう

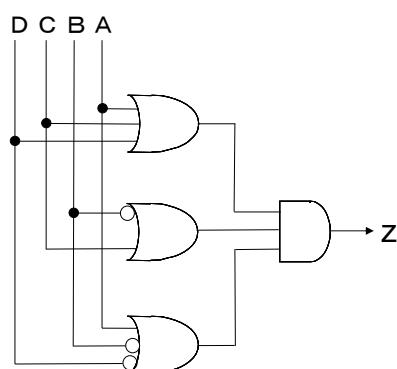
BA DC	00	01	11	10
00		1		
01	1	1	1	1
11	-	-	-	-
10			1	1

次のカルノー図を簡単化しよう

BA	00	01	11	10
DC				
00		1	-	
01		1	-	
11	1		-	-
10		1	-	-

練習問題

次の図の回路について、問い合わせよ。



- ①出力Zについてカルノー図を書き、その簡単化した論理式を示せ。
- ②簡単化した回路を書け。

代表的な組み合わせ回路

- 加算器
 - ハーフアダ – 半加算器 (Half Adder)
 - フルアダ – 全加算機 (Full Adder)
- 減算器
- デコーダ(復号器)
- エンコーダ(符号化器)
- データセレクタ
- コンパレータ

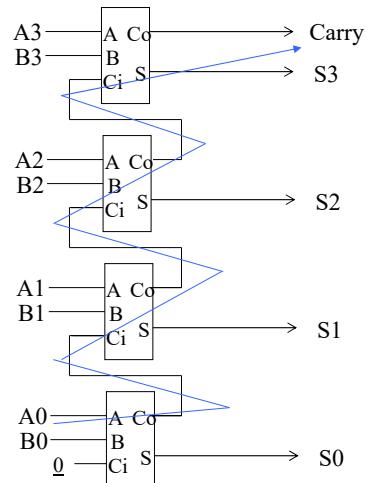
入力の値が同じならば
出力の値も、いつも同じ

加算器(Adder)

- ハーフアダ – 半加算器 (Half Adder)
 - 入力A、Bを受け、その桁の和Sと、桁上げCin (Carry, キャリ)を作り出す2進1桁の加算回路
- フルアダ – 全加算機 (Full Adder)
 - 入力A、Bおよび一つ下の桁からの桁上がりCinを受け、その桁の和Sと、桁上げCoutを作り出す2進1桁の加算回路

Nビットの全加算器

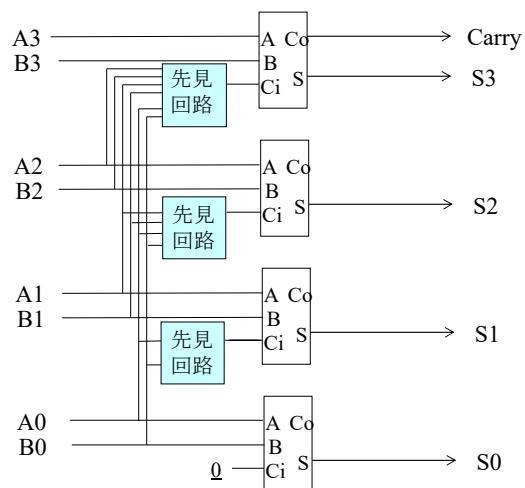
- リップルキャリ方式
 - 桁上げの伝搬
 - 時間がかかる



116

Nビットの全加算器

- キャリルレックアヘッド(桁上げ先見)方式
 - 桁上げを先取り
 - 遅延は小さい
 - 上の桁ほど回路は複雑

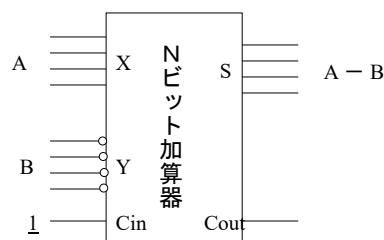


117

減算器

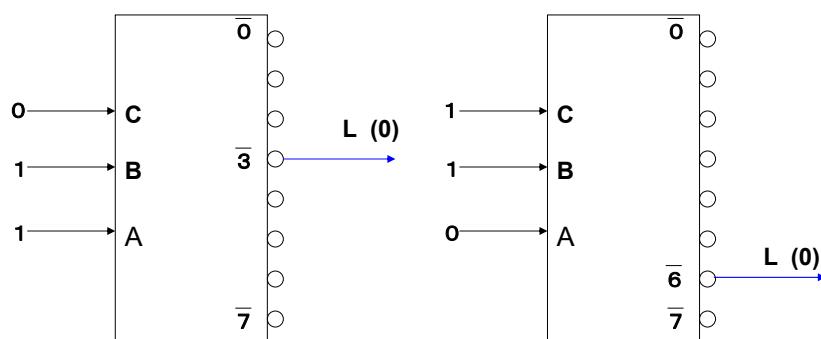
2の補数をつくり加算する
– 引く数を反転して、1を加える

0111	=	7		1111	=	-1
0110	=	6		1110	=	-2
0101	=	5		1101	=	-3
0100	=	4		1100	=	-4
0011	=	3		1011	=	-5
0010	=	2		1010	=	-6
0001	=	1		1001	=	-7
0000	=	0		1000	=	-8



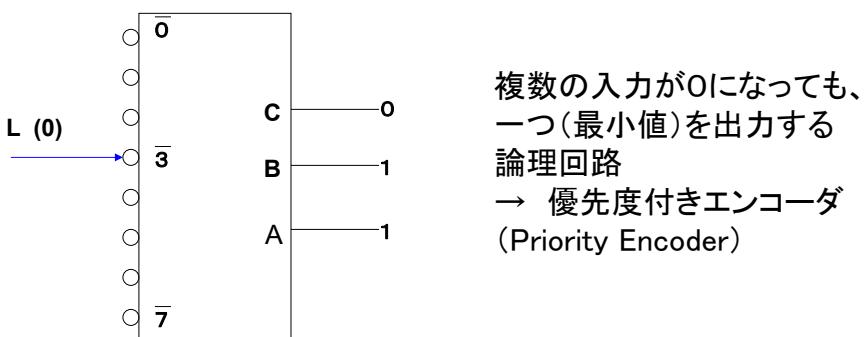
デコーダ(復号器)

- 入力の数値を解読し、どの数がきたのかを示す回路



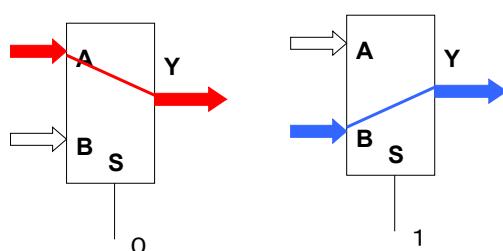
エンコーダ(符号化器)

- アクティブになった入力の数字を符号化して出力
 - i 本目の信号線が0ならば2進符号 i を生成

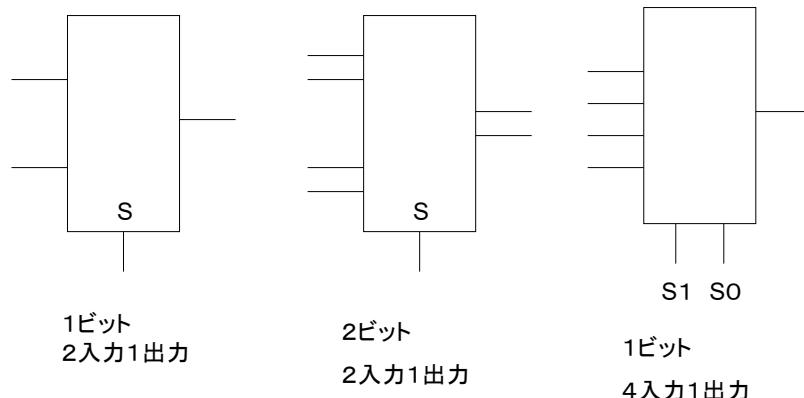


データセレクタ(選択回路)

- 複数の入力線からのデータを選択的に一本の共通線に接続
- マルチプレクサ(multiplexer)とも呼ぶ



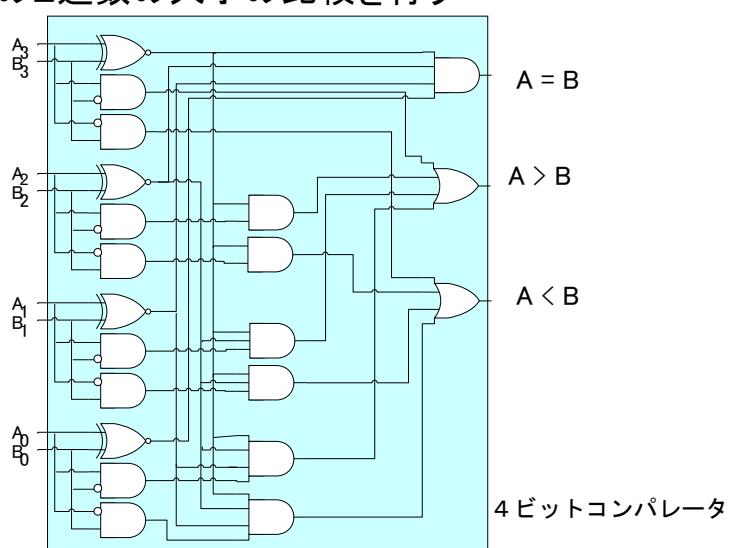
データセレクタのいろいろ



122

コンパレータ(比較器)

- 2つの2進数の大小の比較を行う



123

組み合わせ回路の設計

- 設計すべき機能の把握
- 入力と出力の設定
 - 入力線と出力線の名前付け
- 入出力関係の表記
 - 真理値表その他
- カルノー図などによる最適化
- 簡単化された論理式を導く
- 回路の生成