

集積回路設計 (INTEGRATED CIRCUIT DESIGN) 第9回

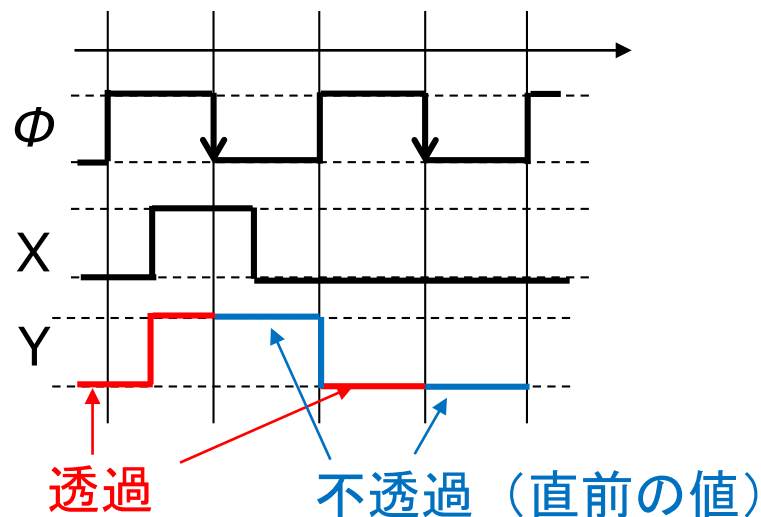
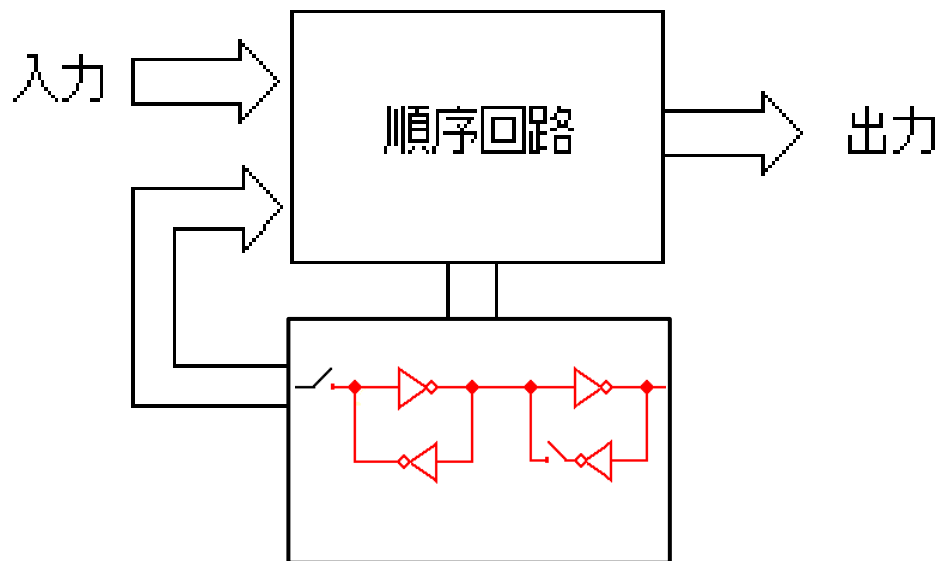
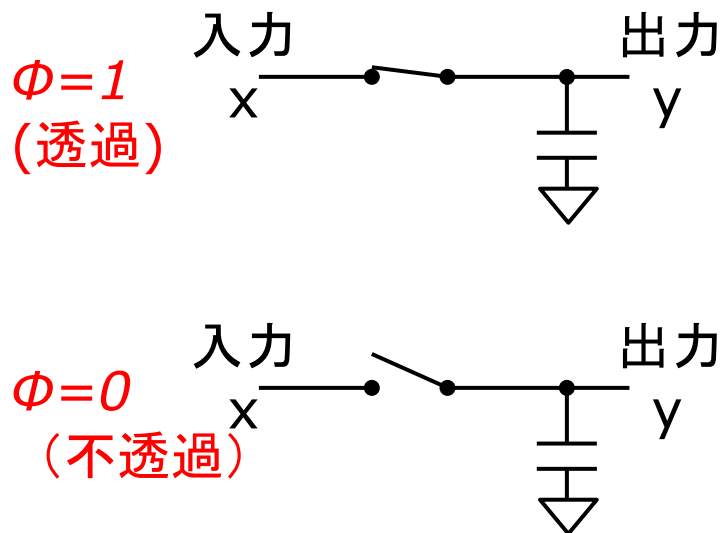
音声が途切れることがあったら、終わるまで
そのままにせずに直ちに教えてください

2020/06/02

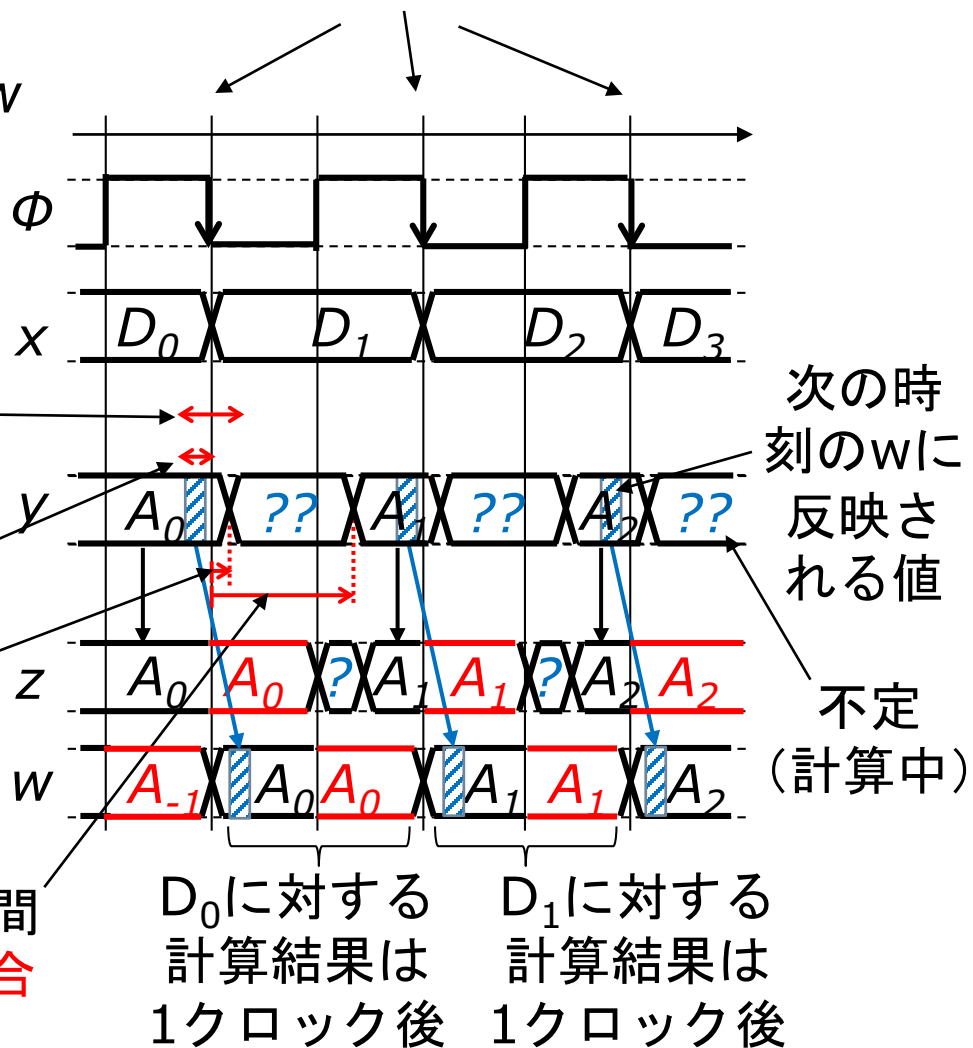
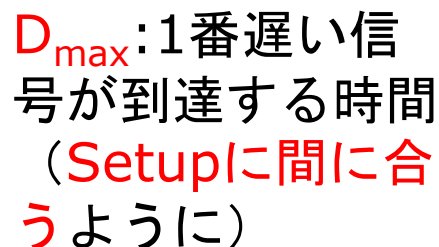
順序回路

7

- メモリを使って順序回路を実現
 - D-FF : 与えられた入力を1周期遅らせて出力から出す



8

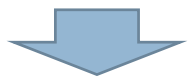


順序回路

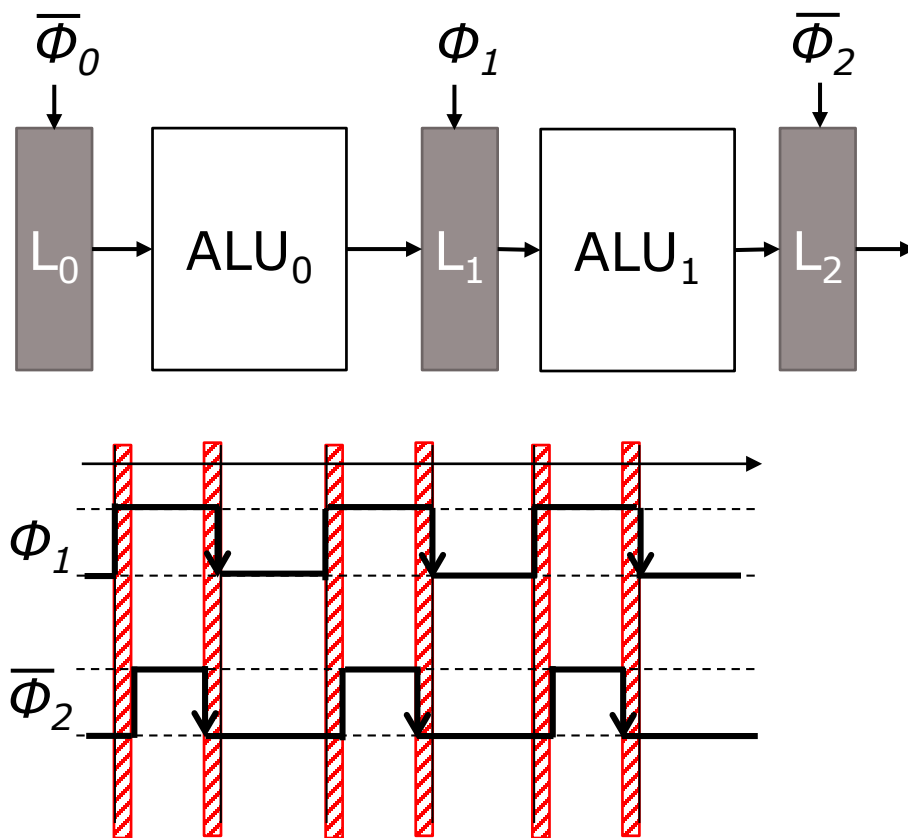
9

□ 2相クロック方式

- 計算の前半と後半をラッチで挟む
- データが突き抜けないように設計



- 両方のラッチがオフする期間を作る
 - このようなクロック分配は設計困難
 - 小規模回路なら可能



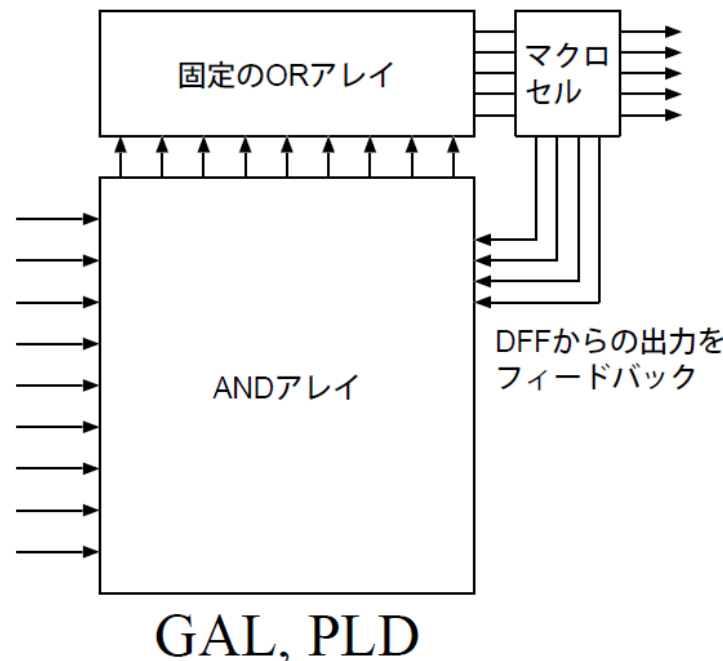
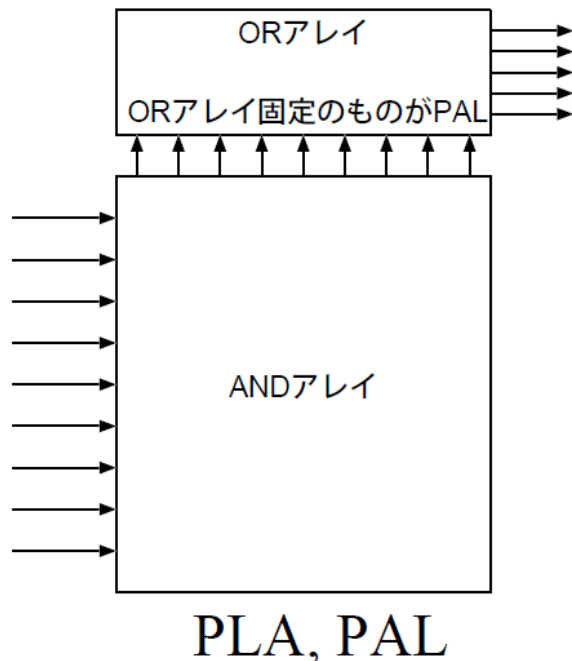
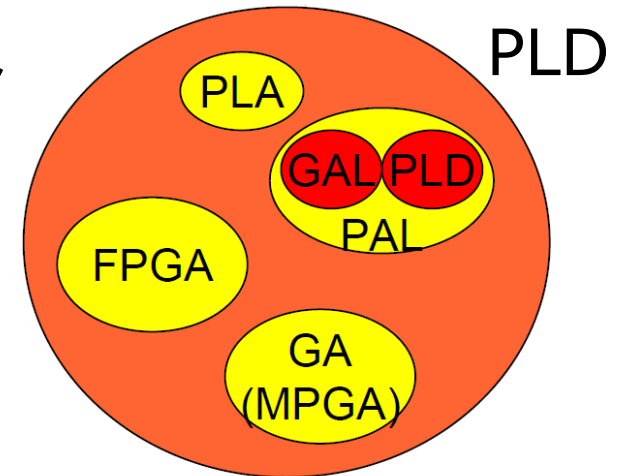
- 広義のPLD (Programmable Logic Device)
 - ▣ 設計者が自由に機能を変更できるLSIの総称
⇔ ASIC (Application Specific IC)
 - ▣ Mask Programmable Device (MPD) : 製造時に
 - ▣ Field Programmable Device (FPD) : その場で

- 狭義のPLD
 - ▣ Programmable Array Logic (PAL)とGeneric Array Logic (GAL)の総称
 - ▣ PAL/PLA : **ANDアレイ・ORアレイ**が書き換え可能
 - ▣ GAL/PLD : **ANDアレイ**は書き換え可能, **ORアレイ**は固定
 - ここのPLDは製品名
 - ORアレイの出力を**ANDアレイ**に☐入力可

PLD

11

- PLA, PAL : 組合せ論理回路のみ
- GAL, PLD : 順序論理回路
 - ▣ 制御回路, カウンタなど

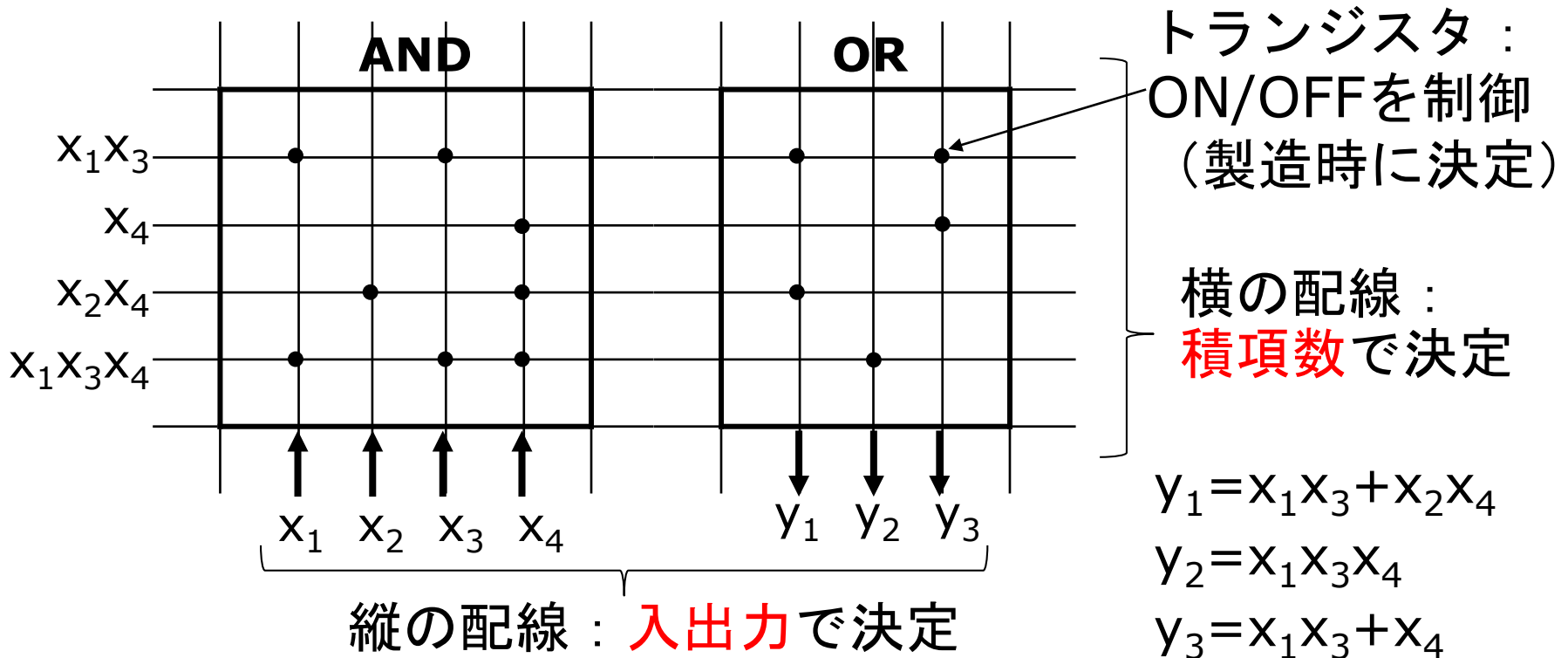


PLD: Programmable Logic Device
PLA: Programmable Logic Array
PAL: Programmable Array Logic
GAL: Generic Array Logic
FPGA: Field Programmable Gate Array
MPGA: Mask Programmable Gate Array
GA: Gate Array

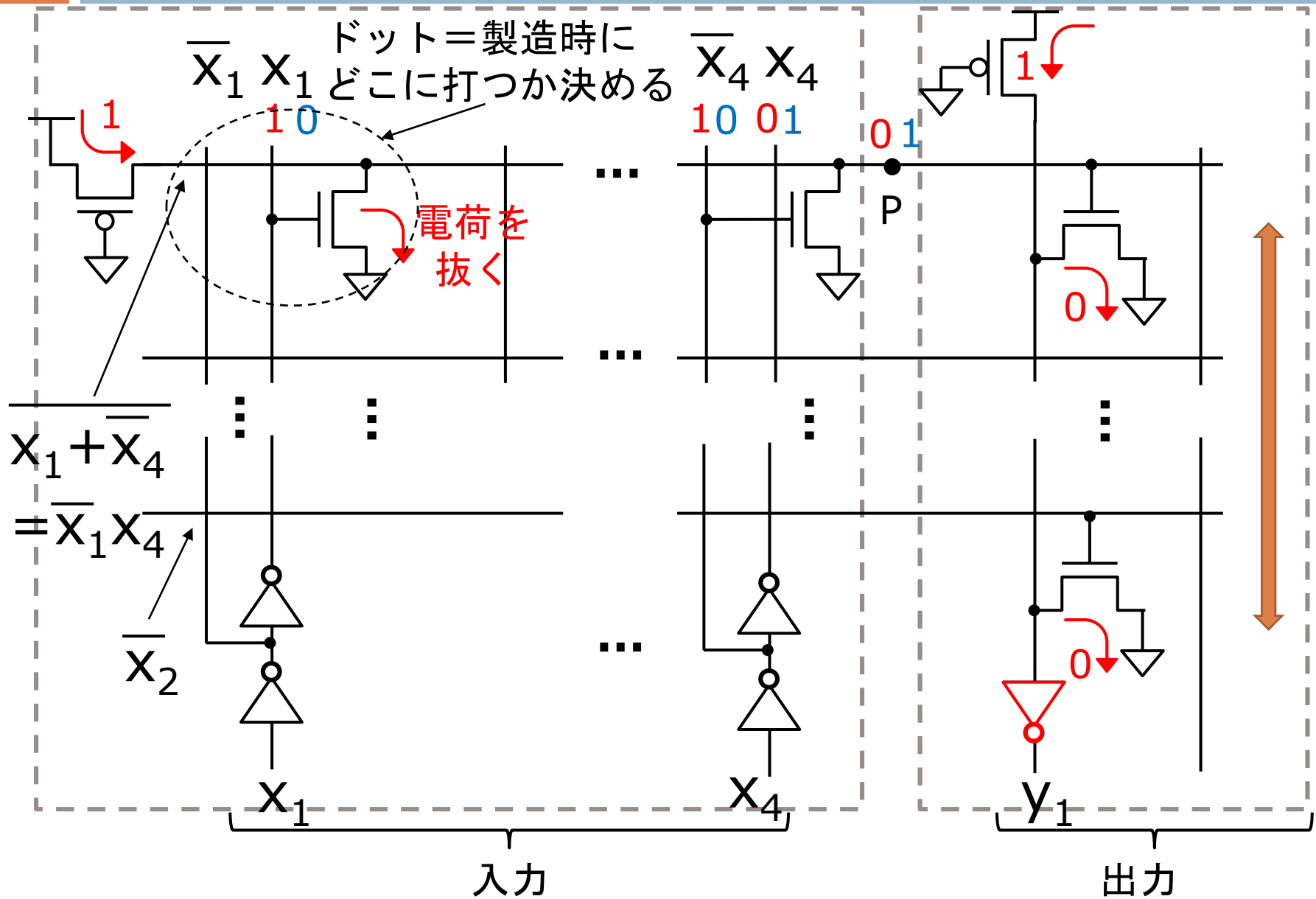
PLA (Programmable Logic Array)

12

- 論理を組む部品（プログラム可能）が配列上に配置
 - ROMの一種
 - 面積：横の配線（積項数）に依存
 - 歩留まり：トランジスタ数（主項の大きさ）に依存

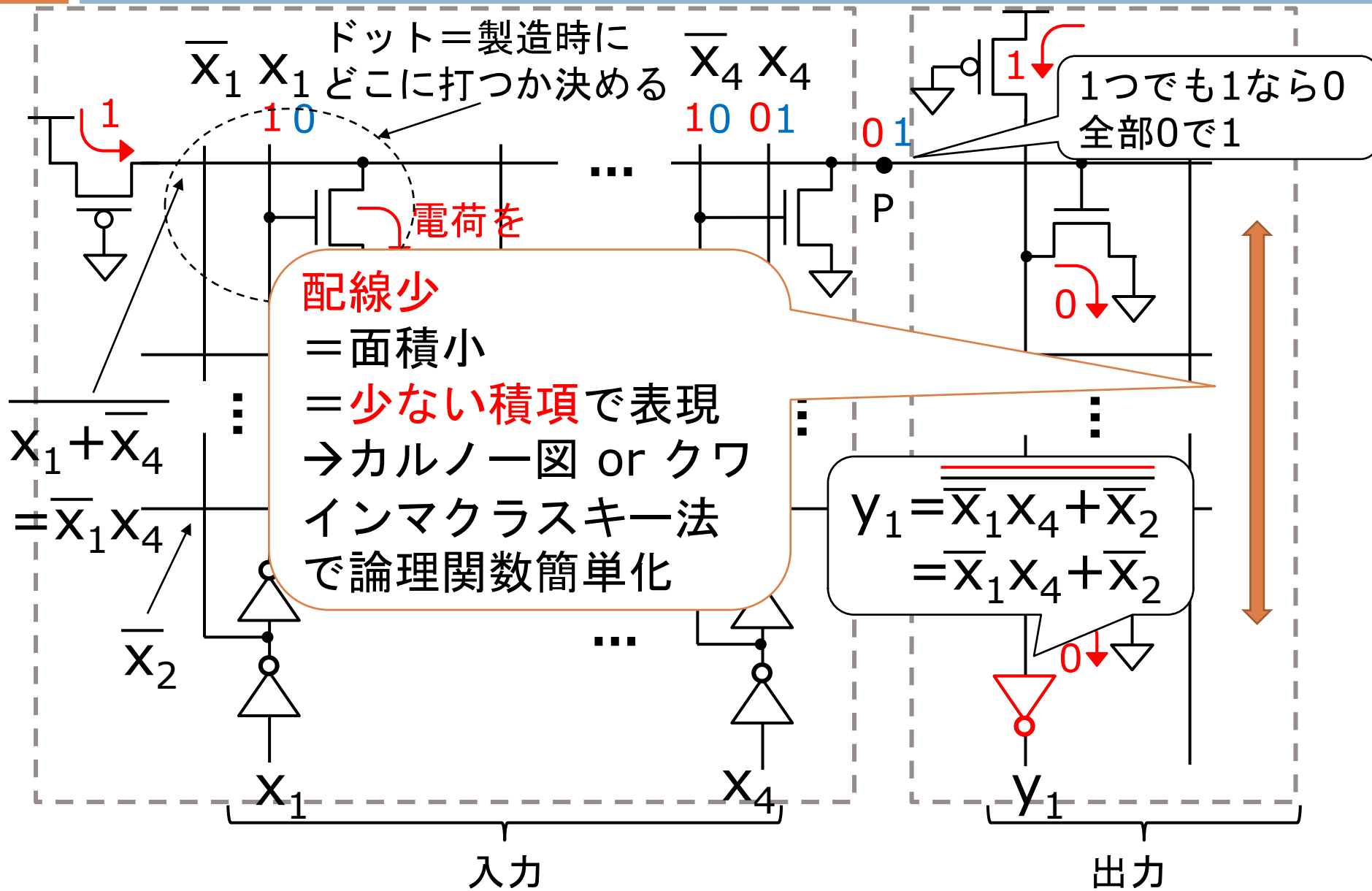


13



PLA (Programmable Logic Array)

14



カルノー図を用いた論理式の簡単化

16

□ $xA \vee \bar{x}A = A$ をもとに*/1をまとめる

▣ 2のべき乗のサイズの枠で囲う



▣ できるだけ少数個の領域で

▣ 1は全て含み、0は全て排除

▣ *は囲みを大きくするために都合の良いものだけ含める

■ 1や*のマスをできるだけ大きく

■ 領域が重なってもOK

4変数の場合

x_2x_1	00	01	11	10
x_4x_3				
00	F			F
01	D	E		
11				
10	F			F

5変数の場合

$x_3x_2x_1$	000	001	011	010	110	111	101	100
x_5x_4								
00		I			J			J
01	G	H				H		
11								
10		I				K		

□ 定まった囲みに対応する
NOT-AND項をORで繋ぐ

クワイン-マクラスキー法

17

- *を1と解釈
- $F=1$ となる入力変数の値の組に対応する極小項を肯定型変数の数でグループ分け
 - p =含まれる肯定型変数の数

$p=1$	0	0	0	1
	0	0	1	0
	1	0	0	0
$p=2$	0	0	1	1
	0	1	0	1
	1	0	0	1
	1	0	1	0
	1	1	0	0
$p=3$	1	1	0	1
$p=4$	1	1	1	1

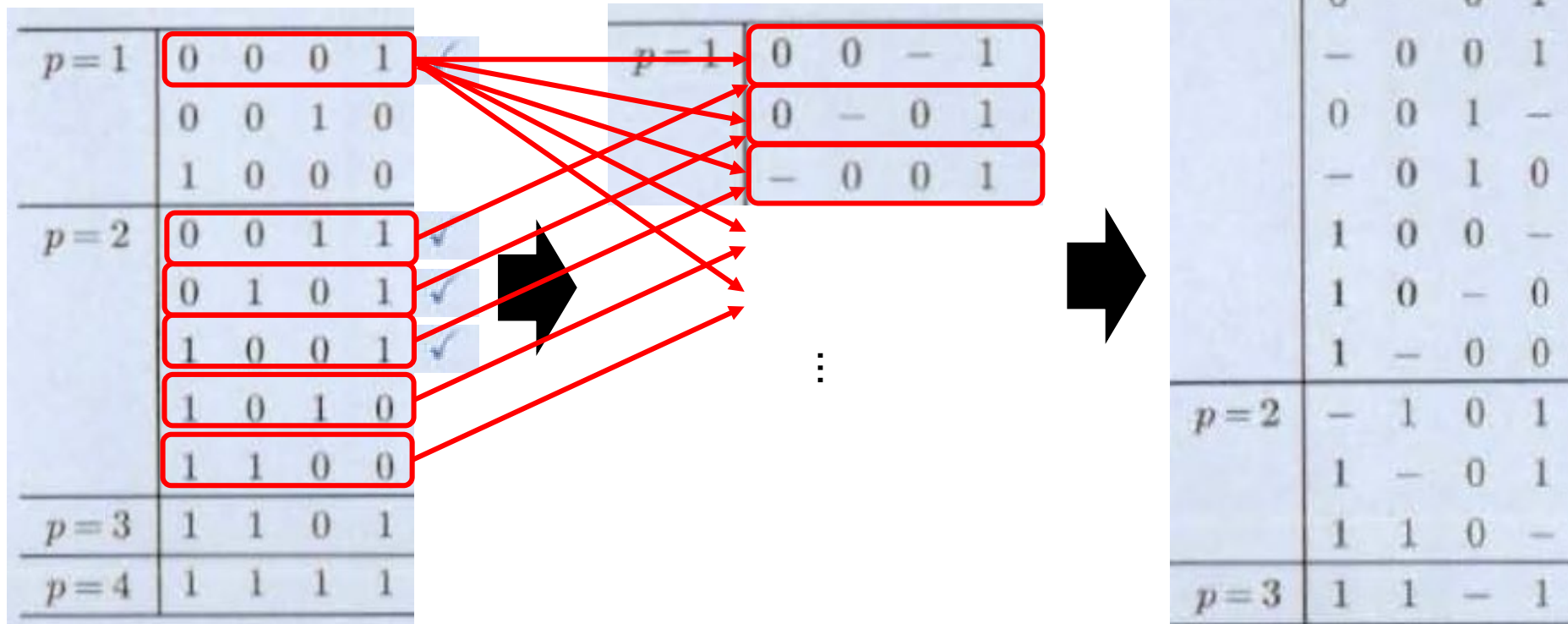
0001:肯定型変数1個

x_4	x_3	x_2	x_1	$F(x_4, x_3, x_2, x_1)$
0	0	0	0	0
0	0	0	1	*
0	0	1	0	1
0	0	1	1	*
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	*
1	0	1	0	1
1	0	1	1	0
1	1	0	0	*
1	1	0	1	*
1	1	1	0	0
1	1	1	1	1

クワイン-マクラスキー法

18

- グループ p に属する極小項 xA に対して、
グループ $p+1$ に属する極小項 $\bar{x}A$ を見つけて**統合**
 - ▣ $p=0$ から始める($\leftarrow 0000$ しか属さない)



クワイン-マクラスキー法

19

- 統合できなくなるまで繰り返す

$p=1$	0	0	-	1	
	0	-	0	1	✓
	-	0	0	1	✓
	0	0	1	-	
	-	0	1	0	
	1	0	0	-	✓
	1	0	-	0	
$p=2$	1	-	0	0	✓
	-	1	0	1	✓
	1	-	0	1	✓
	1	1	0	-	✓
$p=3$	1	1	-	1	



$p=1$	-	-	0	1
	1	-	0	-

チェックが入っている項
=他のNOT-AND項で表現されている



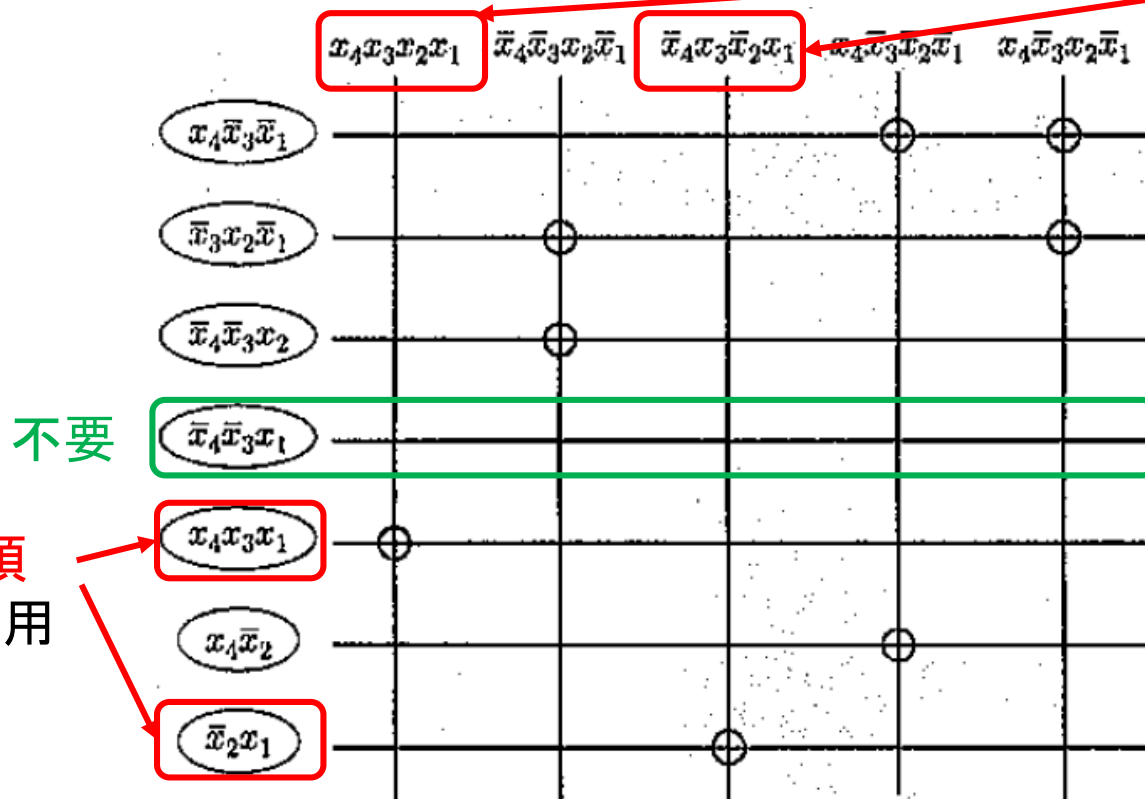
チェックが入っていない
NOT-AND項=主項

包含図

20

- *を0と解釈して**包含図**を構成
 - ▣ 包含関係の意味で、できるだけ**大きな主項**
 - ▣ できるだけ**少ない数**の主項

これらの極小項を
包含する主項は
それぞれ**1つだけ**

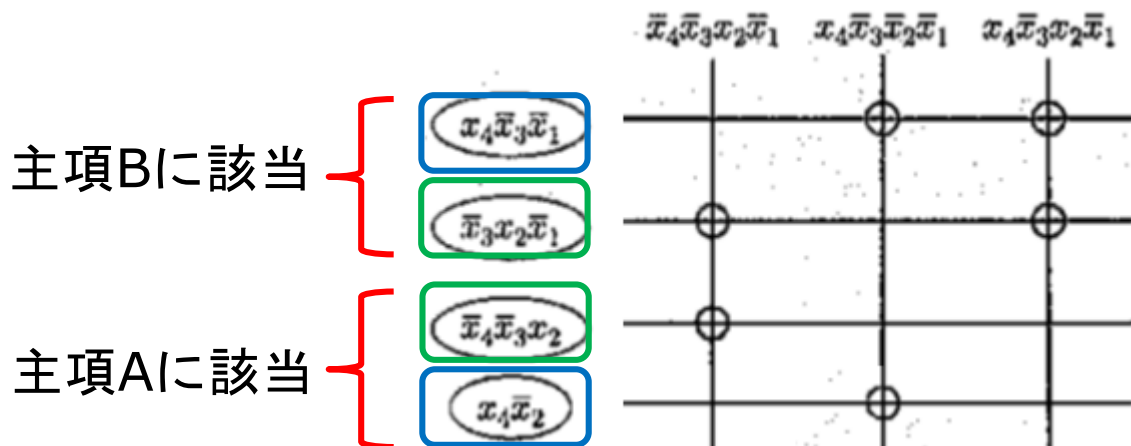


① 使用が確定した主項と、その主項が**包含する極小項**は、検討対象から外して考える

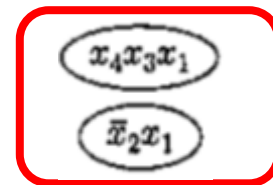
包含図

21

□ 包含図を小さくして繰り返す



使用が確定
した主項



② ある主項Aが包含する極小項が、全て別の主項Bでも包含
= BがAを代行
→ **A**を包含図から**除外**
(= **B**を**選択**)

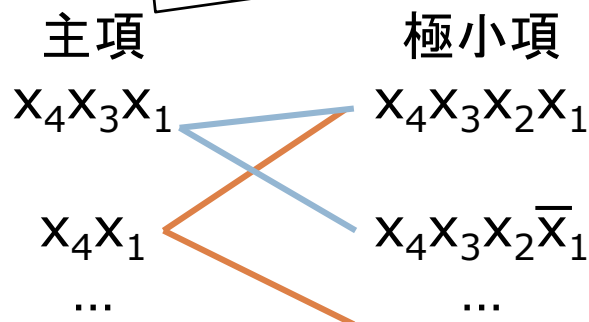
□ 総当たり（全数探索）

▣ 確実だが計算量は膨大

□ ヒューリスティック

▣ Minimum Cover Problem
→ 二分グラフ

すべての極小項をカバーする
最少・より上位の主項を選択

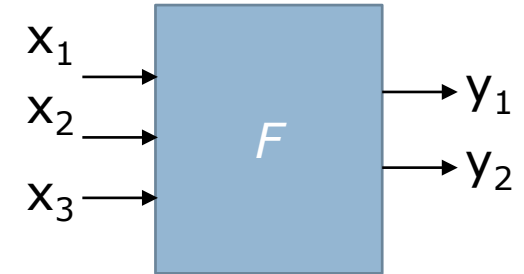


複数出力の論理回路

22

複数論理関数の同時
簡単化=PLA設計

- 各出力について簡単化するより、**共通部分**を使って簡単化した方が論理素子数を減らすチャンス大



- 共用できる**極小項のみ

x_2x_1	00	01	11	10
x_3				
0			1	
1		1	*	

$\rightarrow x_3x_1, x_2x_1$

- 共用できない**極小項を補足

y_1

x_2x_1	00	01	11	10
x_3				
0	1		1	*
1		1	*	

$\rightarrow x'_3x'_1, x'_3x_2, x_3x_1, x_2x_1$

y_2

x_2x_1	00	01	11	10
x_3				
0			1	
1	*	*	*	1

$\rightarrow x_3, x_2x_1$

x_3	x_2	x_1	y_1	y_2
0	0	0	1	0
0	0	1	0	0
0	1	0	*	0
0	1	1	1	1
1	0	0	0	*
1	0	1	1	*
1	1	0	0	1
1	1	1	*	*

単一出力と同様に
ドントケアは1と
して極小項を抽出

複数出力の論理回路

23

- 包含図を用いて主項を選択
 - ▣ y_1 用と y_2 用を混同しないように
 - ▣ 共用の主項を無理に使う必要はない

