

第5章 標準ロジックICの 機能と使い方

5-1 組み合わせ論理回路

組み合わせ論理回路とは、出力の論理状態がその時点で入力されている入力論理状態でのみ決定される論理回路です。つまり、記憶ループをもっていないので、入力と出力の論理関係が1対1で、その時点で決まります。インバータ(NOT)、OR、ANDの三つの基本論理回路さえあれば、すべての組み合わせ論理回路が実現できます。

	AND	NAND	OR	NOR	ExOR	ExNOR	INV	
正論理								
負論理								
論理式	$Y = A \cdot B$	$Y = \overline{A \cdot B}$	$Y = A + B$	$Y = \overline{A + B}$	$Y = (A+B) \cdot (\overline{A+B})$	$Y = (A \cdot B) + (\overline{A \cdot B})$	$Y = \overline{A}$	
真理値表	入力	出力						
	A B	Y						
	L L	L	H	L	H	L	H	H
	L H	L	H	H	L	H	L	H
	H L	L	H	H	L	H	L	L
H H	H	L	H	L	L	H	L	

図5.1 ゲートの論理記号と論理式と真理値表

● ゲートとは

ゲートは、ブール代数(論理式)や2値論理記号で表される論理演算の最小単位がICになった製品です。図5.1に、ゲートの論理記号と論理式、その真理値表を示します。論理記号の表記方法には正論理と負論理がありますが、最近では正論理だけを用いるようになってきました。CAD(Computer Aided Design)を用いてコンピュータが自動的に論理設計を行う時代になり、CADツールに登録されるロジック・シンボルは正論理だけで事足りるようになったからです。

ゲートには、

- (1) 信号を単に反転するINV(インバータ)
- (2) 論理はそのままで、波形整形や信号のタイミングをとるBuffer(バッファ)
- (3) 基本の演算回路であるAND(アンド), OR(オア), ExOR(イクスクルーシブ)

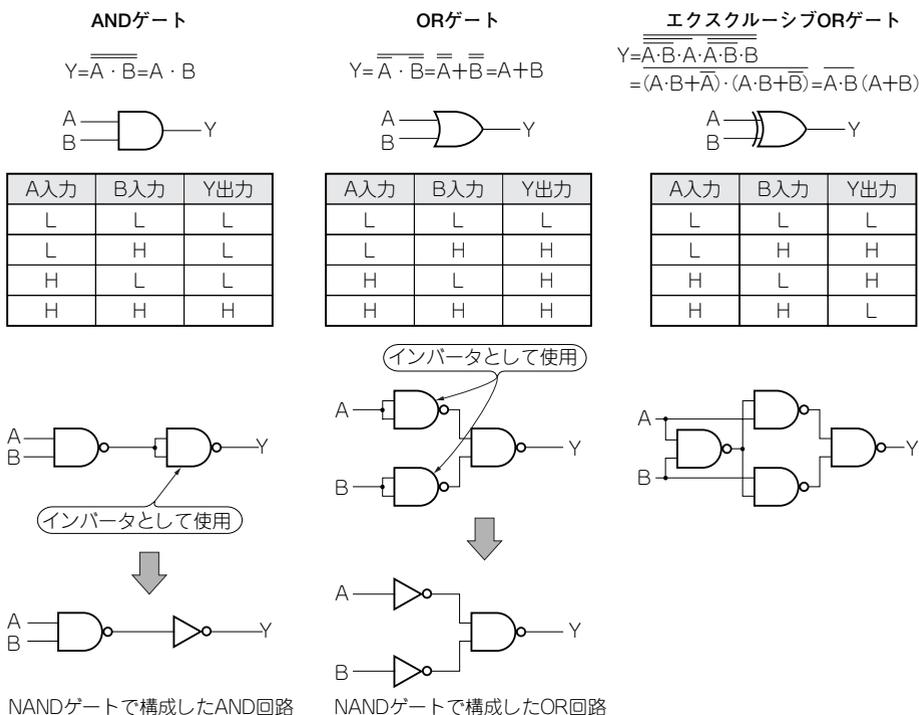


図5.2 NANDゲートを用いたゲート変換の例

ブ・オア)

(4) 基本の演算回路とインバータを組み合わせ、論理を反転したNAND(ナンド), NOR(ノア), Ex-NOR(イクスクルーシブ・ノア)があります。

● ゲートの応用例

(1) ゲート機能の変換

NANDゲートを用いたゲート変換の例を図5.2に、NORゲートを用いたゲート変換の例を図5.3に示します。74HC00, 74VHC00, 74LCX00など、シリーズ名の後に続く数字“00”は、2入力NANDを定義しており、標準ロジックでは図5.4のように、通常、一つのパッケージに四つの2入力NANDが入っています(ワ

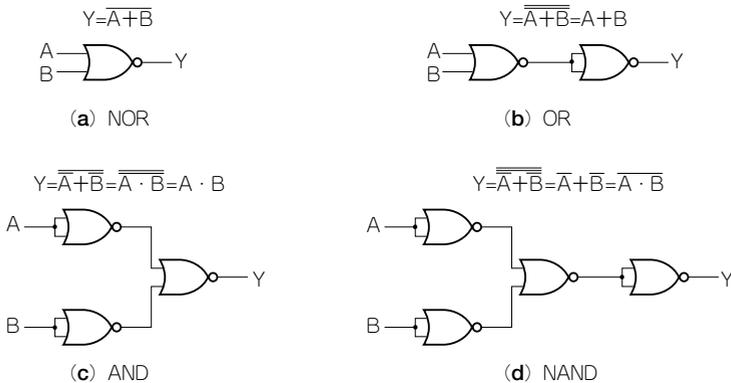


図5.3 NORゲートを用いたゲート変換の例

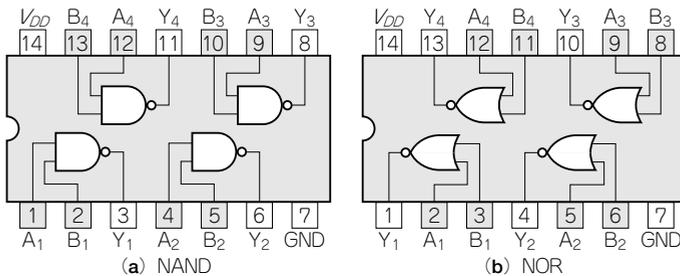


図5.4 4回路入り2入力NANDゲートとNORゲート

ンゲート・ロジックは別)。2入力NORの場合は“02”であり、同じく一つのパッケージに4回路が入っています。一つのデバイスを元にいろいろと実験することができます。

(2) ゲート入力の拡張法

図5.1では、入力が1本または2本の最少論理単位を示しましたが、入力本数を増やす方法を図5.5に示します。基本構成が2入力1出力であっても、論理変換を行えば容易に複数入力の論理回路を構成できます。

(3) CR発振回路への応用

a) 2入力NANDゲートの例

発振制御付きの3段発振回路を組んだ例を図5.6に示します。

この回路は、正帰還の後に負帰還ループを組んだ回路構成になっています。今、

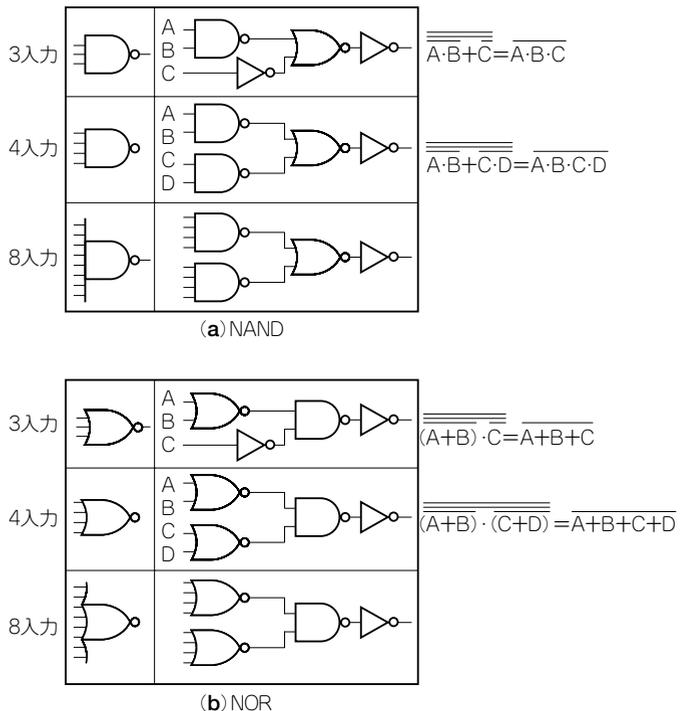


図5.5
多入力ゲートの構成

制御入力 A_1 が“H”レベルのとき、コンデンサ C の電荷は初めゼロです。そして、④点の電位を GND とすると、③点と②点、①点の電位はそれぞれ GND, GND, V_{DD} となります。3段目のゲートは抵抗 R を通してコンデンサ C を充電するので、④点の電位は徐々に上がります。そして、1段目のゲートのスレッショルド電圧 V_{TH} を超えると、各段のゲートは反転します。

そのため、④点の電位は $(V_{TH} + V_{DD})$ 、③点は GND となり、コンデンサ C の電荷は $C \cdot (V_{TH} + V_{DD})$ となります。そして、次にはコンデンサ C の電荷は抵抗 R を通して放電し、④点の電位が1段目のゲートのスレッショルド電圧 V_{TH} をふたたび超えたときに、各ゲートはふたたび反転します。

この動作により、④点の電位は $(V_{TH} - V_{DD})$ 、③点は V_{DD} となり、コンデンサ C に充電を始めます。以上の動作の繰り返しにより、発振を継続します。

A_1 が“L”レベルのときは、③、②、①、④点の電位は、それぞれ V_{DD} , V_{DD} , GND, GND となり、状態が強制的に安定するため発振しません。

なお、このような発振回路は、1段目の入力に V_{DD} より高く、GND より低い電圧が加わります。したがって、過電圧の印加で流れる入力電流でデバイスが破壊・劣化する可能性があるため、かならず入力保護抵抗 R_p を入れる必要があります。

ただし、入力保護抵抗の値が大きすぎるとこの抵抗にノイズが乗り、出力の反転時にチャタリング(不要なヒゲ状信号)を生じることがあります。逆に、抵抗 R_p が小さいと、入力端子に過電流が流れるため、入力保護としての用をなしません。

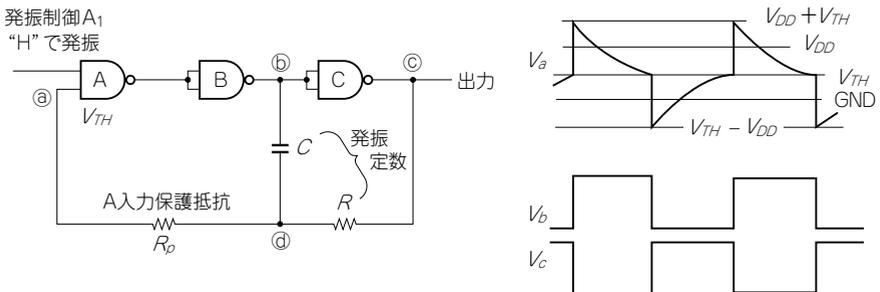


図5.6 2入力NAND3段CR発振回路

入力保護抵抗は、数kΩ～数十kΩ程度が適当でしょう。
 発振周波数の時定数は、次の計算式で目安を出せます。

$$T = 2.2CR = \frac{1}{f_{osc}}$$

図5.7に、発振定数を決定する抵抗 R 、容量 C に対する、発振周波数の傾向を示します。高い周波数で特性が頭打ちになるのは、発振回路を構成するCMOS ICの応答性/伝搬遅延時間の性能によります。

b) 2入力NORゲートの例

2入力NANDゲートの例では、“L”で発振停止となるコントロール端子がありました。図5.8では2入力NORゲートを用い、“H”で発振を停止させることができます。発振のメカニズムや特性に差はありません。

(4) 単安定マルチバイブレータへの応用

2入力NORゲートを用いた単安定マルチバイブレータ(モノステーブル・マルチバイブレータ、略称モノマルチ)への応用を考えてみましょう。

モノマルチとは、入力される論理信号の立ち上がり、または立ち下がりのエッジをとらえて所要の時間幅をもった出力パルスを取り出す回路です。後で説明するマルチバイブレータはこれをIC化したものです。精度を要するシステムへの応用は、IC化された製品を用いることをお勧めします。

図5.9に示すように、各種ゲートICを用いてモノマルチを構成できますが、ここでは図5.10 [図5.9(a)] を例にして説明します。

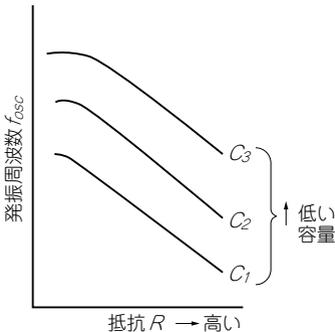


図5.7 発振周波数の特性

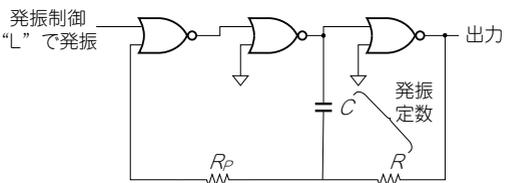


図5.8 発振制御の例

NORゲートで構成されたこの回路は、入力の立ち上がりエッジで出力に“H”レベルのパルスが出力されます。出力パルス幅は C_T と R_T の時定数で決定され、

$$V_a = V_{DD} [1 - \exp(-t/R_T C_T)]$$

より、

$$t = -R_T C_T \cdot \ln [1 - (V_a/V_{DD})]$$

したがって、 V_{TH2} をNORゲートIC₂のスレッシュド電圧とすると、

$$t_w = -R_T C_T \cdot \ln [1 - (V_{TH2}/V_{DD})]$$

で表せます。

ここで、 $V_{TH2} = (1/2)V_{DD}$ とすると出力パルス幅 t_w は、

$$t_w = 0.69 C_T R_T$$

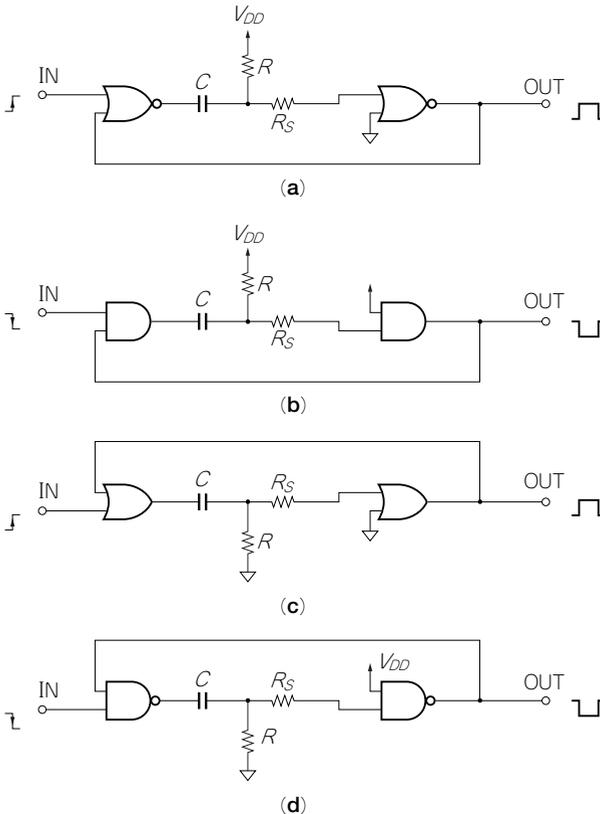


図5.9 各種ゲートICを用いた単安定マルチバイブレータ

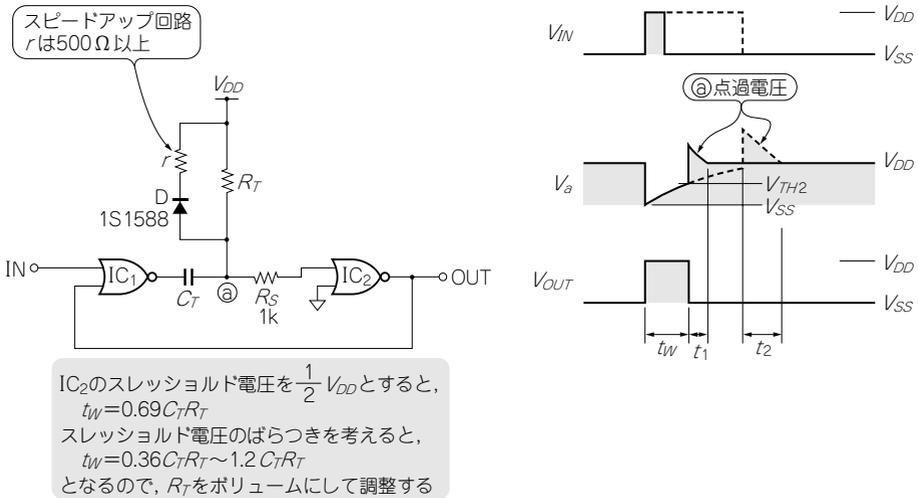


図5.10 単安定マルチバイブレータの動作

仮に V_{TH2} が $0.3V_{DD} \sim 0.7V_{DD}$ の最悪レベルまでばらつくとする、

$$t_w = 0.36C_T R_T \sim 1.2C_T R_T$$

の範囲でばらつきます。したがって、出力パルス幅を規定時間に入れたい場合には、 R_T を可変抵抗と固定抵抗の併用にする必要があります。

R_S は、㊸点の過電圧に対する入力保護抵抗で、1kΩ程度でよいでしょう。

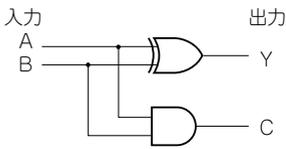
また、図5.10において、トリガ・パルス幅が出力パルス幅より短い場合は t_1 、トリガ・パルス幅が出力パルス幅より長い場合は t_2 で示すリカバリ時間を要するため、このリカバリ時間を短くする必要があるときには、図に示すダイオード+抵抗 (D+r) のスピードアップ回路を挿入します。rの値は数百Ω以上を推奨します。

(5) 加減算回路への応用

二つの信号を単純に加算する回路を図5.11に示します。

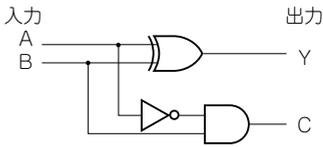
AとBが入力で、Yは加算された出力、Cはキャリ出力となります。真理値表に示すように、AとBが“1”のときキャリ出力Cが“1”になります。

また、減算器は図5.12で構成できます。これら二つの回路例は、ともに半加算器、あるいは半減算器と呼ばれます。半加算器は、前段からの桁上げ加算機能が考慮されていない組み合わせ論理回路であり、半減算器は前段からの桁借り減



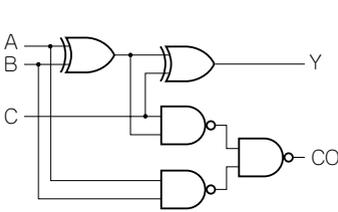
入力		出力	
A	B	Y	C
0	0	0	0
1	0	1	0
0	1	1	0
1	1	0	1

図5.11 半加算器



入力		出力	
A	B	Y	C
0	0	0	0
1	0	1	0
0	1	1	1
1	1	0	0

図5.12 半減算器



入力			出力	
A	B	C	Y	CO
0	0	0	0	0
1	0	0	1	0
0	1	0	1	0
1	1	0	0	1
0	0	1	1	0
1	0	1	0	1
0	1	1	0	1
1	1	1	1	1

A: 被演算数
 B: 演算数
 C: 前ビットからの桁上げ
 Y: 演算によって得られる和
 CO: 桁上げ

図5.13 全加算器

算機能が考慮されていない組み合わせ論理回路です。

次に、前ビットからの桁上げを考慮した全加算器，および桁借りを考慮した全減算器を考えてみると図5.13，図5.14のようになります。

全加算器，全減算器では，それぞれ桁上げ(キャリ：CO)出力と，桁借(ボロウ：BO)出力が備わっており，図5.13ではA, B, Cどの入力信号でも“H”(“1”)→“L”(“0”)に桁上げがあるとき，COに“H”(“1”)が出力されます。また，図5.14ではA, B, Cのどの入力信号でも“L”(“0”)→“H”(“1”)に桁借りがあるとき，BOに“H”(“1”)が出力されます。