

§ 9. コンピュータ回路の基礎—レジスタと加算器—

コンピュータの基本機能：（一時的）記憶と演算

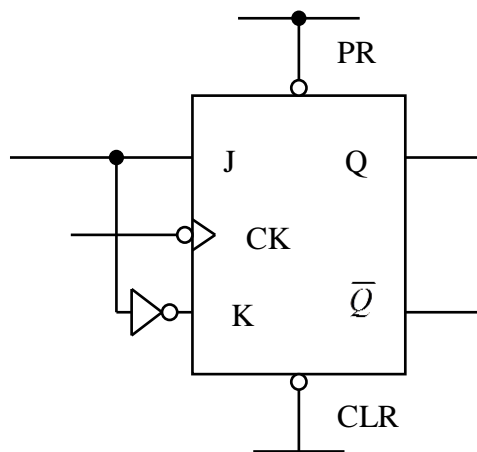
例：電卓で行う演算 $A \leftarrow A + B$

A は電卓の表示部分：数を一時的に置く機能（レジスタ）が必須

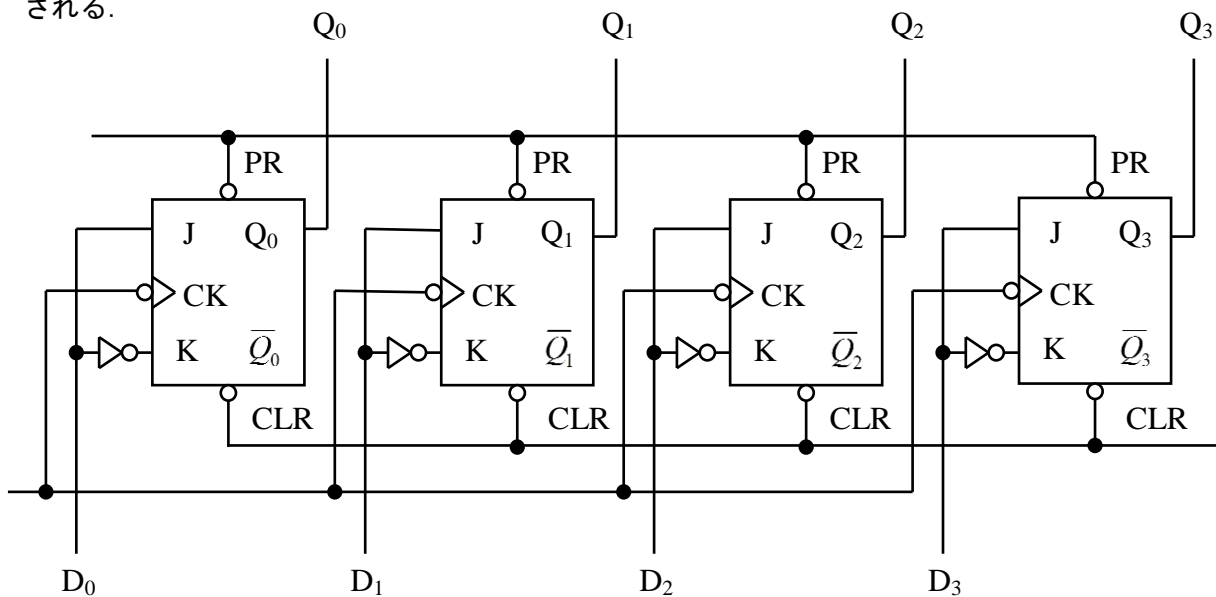
<レジスタ（register：置数器）>

コンピュータ（CPU）の内部にある，データやアドレスなどを一時的に保存しておく記憶装置
外部メモリよりも高速にアクセスできる

JK-FF で作ることができる：1 bit の記憶ができる。



4 bit のレジスタ： $D_3D_2D_1D_0$ の 4 ビットデータが，クロックパルスで記憶され， $Q_3Q_2Q_1Q_0$ に出力される。



<加算器>

演算：論理演算と算術演算

論理演算 (AND, OR, NOT) は, ブール代数にしたがって行えば良い.

算術演算は論理演算とは異なる (例: 論理演算では $1+1=1$ だが, 算術演算では $1+1=10_{(2)}$)

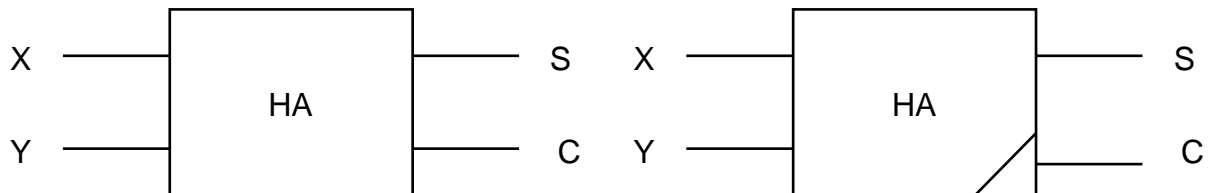
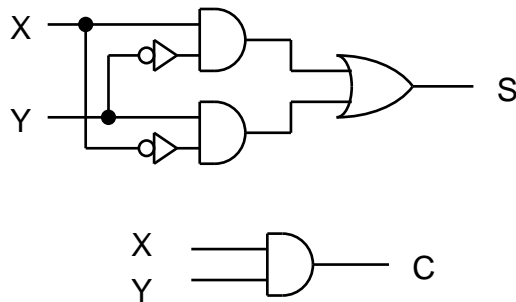
算術演算 (四則演算: 加減乗除)

減算は負の数 (補数) を加える. 乗算は桁ずらし加算. 除算は難しい (特別なルーチンが必要).

1 桁の加算:

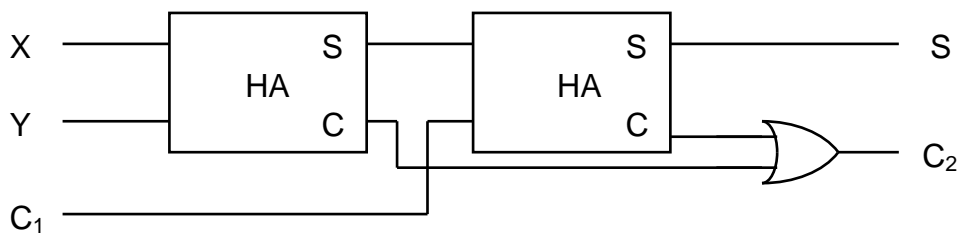
$X + Y = S$, これと桁上がり (キャリー: C)

X	Y	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

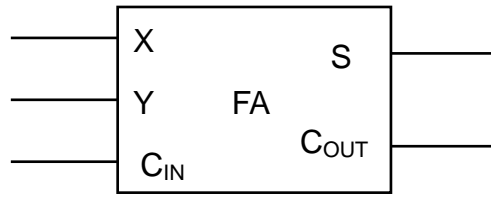


二つの入力 (X と Y) に対し, その和 (S) と桁上がり (C) を出力する回路を, 半加算器 (Half Adder) と呼び, 上のような記号で書く. ただし, これは, 下の桁からの桁上がりの処理ができないため, 半加算器を 2 個使用した全加算器 (Full Adder) を用いる.

$X + Y + C_1 \rightarrow S, C_2$

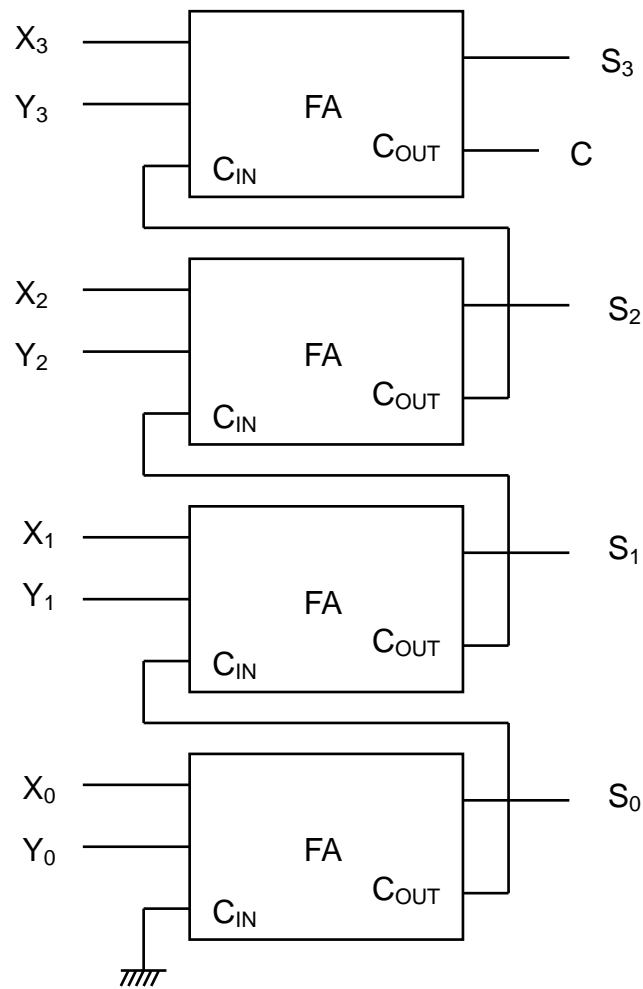


半加算器による全加算器の構成



全加算器の記号の例（キャリー入力とキャリー出力がある）

○4ビットの加算器



$$\begin{array}{r}
 X_3 X_2 X_1 X_0 \\
 +) Y_3 Y_2 Y_1 Y_0 \\
 \hline
 C S_3 S_2 S_1 S_0
 \end{array}$$