H29 ②

以下記法は全てMIL記法に基づくものとする。

(1) 真理値表は以下。

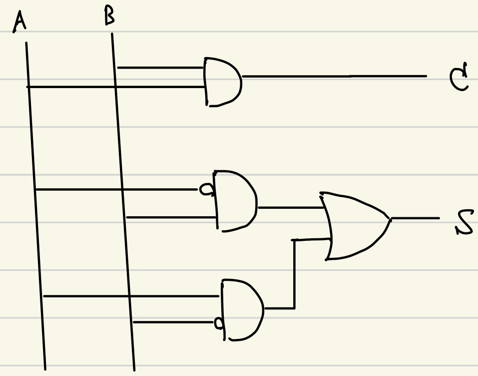
A	B	c	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

(2) (1)より

$$\begin{cases} c = AB \\ S = A \oplus B \ (= A\ \mathrm{XOR}\ B) \end{cases}$$

$= A\bar{B} + \bar{A}B$ になるため。

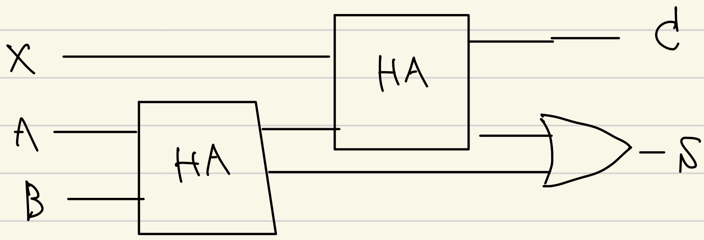
HAの回路は以下の通り



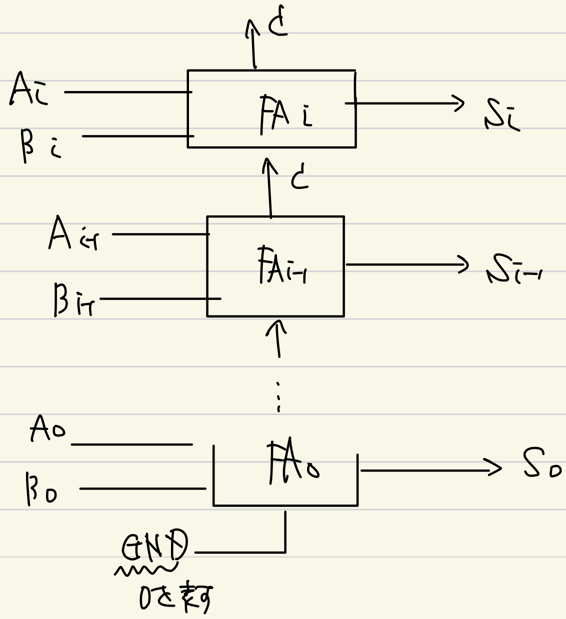
(3) 真理値表は以下

X	A	B	c	S
0	0	0		0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

(4) FAの回路は以下の通り。



(5) nコのFAを直列に連結して、i番目のFAでは2つの数のi bit目と、i-1コ目のFAからの桁上がりを入力として受けとる。その連結の様子を以下に図として示す。



(但し、A_i, B_i はそれぞれ入力された2数のi bit目の値, S_i は A+B のi bit目の値を表すものとする。)

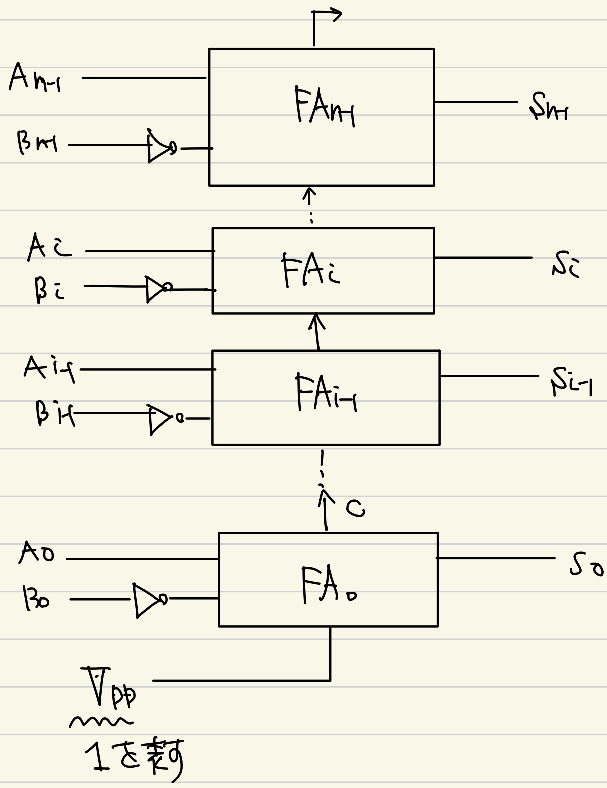
この方式はカスケード式接続と呼ばれることがある。

(6) (5)の方式では、高位のFAにおける桁上がり信号の生成の遅延がNに比例して大きくなるため、規模の大きい加算器に向かないという欠点がある。この欠点を解消するため、
$A_0, \sim A_{i-1}$, $B_0, \cdots B_{i-1}$ から直接 FA_i に入る桁上がり信号を生成するような論理回路を設計する方法がとれる。これをキャリールックアヘッド方式などと呼ぶ。
この方式は、カスケード方式と比較して多くの素子を要するが、遅延のオーダを $\log N$ に比例する値に収えることができる。

(7) nbit 符号付き整数 A, B を入力にもち、A−B を計算する回路を考える。
この時、$*B$ で Bの補数を表すものとすると、

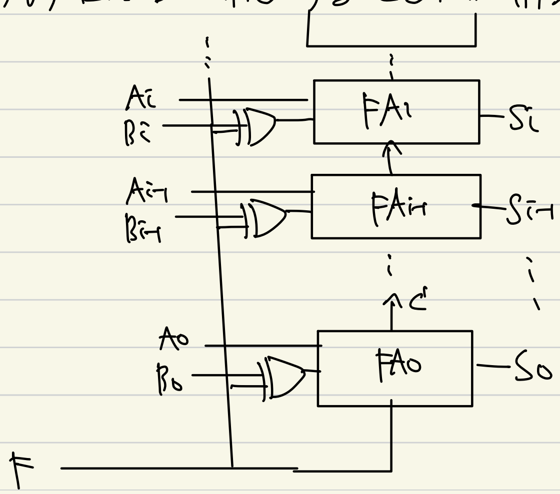
$$A - B = A + {*B} = A + (2^n - B) = A + (2^n - 1 - B) + 1$$
$$= A + (\text{Bの全bitを反転した数}) + 1$$

となるから、(5)に用いたカスケード方式の回路を以下のように書き換えることで、減算器を作ることができる。



(但し、入出力に関する表記は(5)と共通)

(8) (5)(7)の回路を統一することで、加算/減算選択付き回路は以下のように表せる。



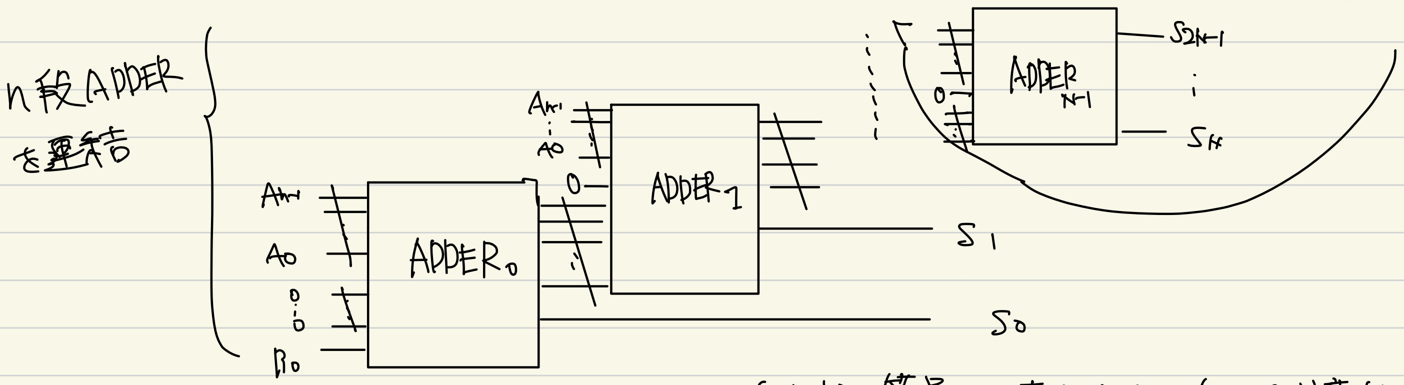
(但し、入出力に関する表記は(5)(7)と共通であり、F=0の場合加算器、F=1の場合減算器として動作するものとする)

(9) $\begin{cases} A = A_{n-1}A_{n-2}\cdots A_0 \\ B = B_{n-1}B_{n-2}\cdots B_0 \end{cases}$ と表すと

$$A\times B = (A_{n-1}\cdots A_0 \cdot B_0) + (A_{n-1}\cdots A_0 \times B_{n-1}\cdots B_1) \ll 1$$

と表記できることを

利用して、nコの全加算器を連結することで得られる乗算器を構成すると以下のように書ける。



(nbit 符号なし整数を入力にもつことに注意する)

なお、簡単のため加算器ADDERをブラックボックスとして使用している。(インターフェースは(a)に準ずる)