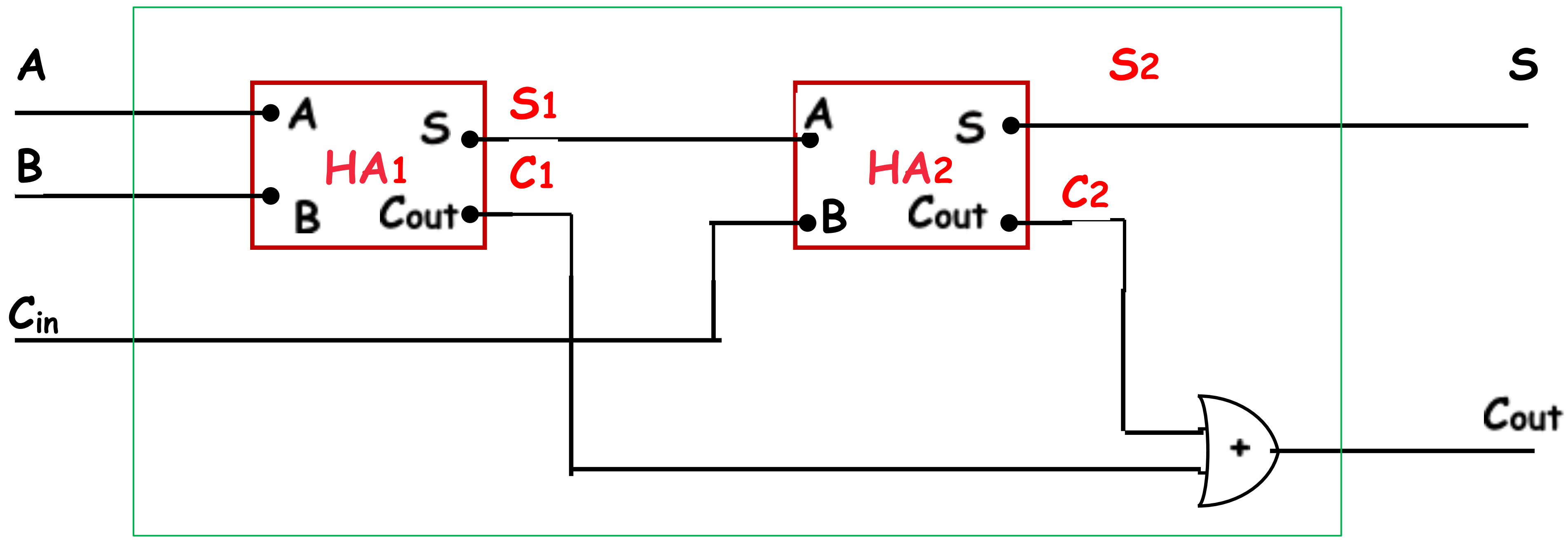
# Half Full Subtractor

## CLASS 17

# HW 17.4 Construct a FA using only HA's and one other gate.

## Solution

Design: FA $\begin{array}{r} C_{out} \nwarrow \\ A + \\ B \\ C_{in} \\ \hline S \end{array}$ using: HA1 $\begin{array}{r} C_1 \nwarrow \\ A + \\ B \\ \hline S_1 \end{array}$ HA2 $\begin{array}{r} C_2 \nwarrow \\ S_1 + \\ C_{in} \\ \hline S_2 \end{array}$




**Note**: We would think we need to add $C_1$ and $C_2$. Do we? Not if they may not be both =1. Let's see:

Suppose $C_1 = 1 \dashrightarrow \left\{\begin{array}{l} A = 1 \\ \& \\ B = 1 \end{array}\right\} \dashrightarrow S_1 = 0 \dashrightarrow C_2 = 0$ So: $C_1$ and $C_2$ may not be both 1 ---> use OR gate.

We can also prove the diagram is correct by substituting the functions in the diagram:

**HA**: $S = A'B + AB'$
$C = AB$

**HA1**: $S_1 = A'B + AB'$ ; $C_1 = AB$

**HA2**: $S_2 = S'_1 C_{in} + S_1 C_{in}' = (A'B + AB')' C_{in} + (A'B + AB') C_{in}' = \dots$

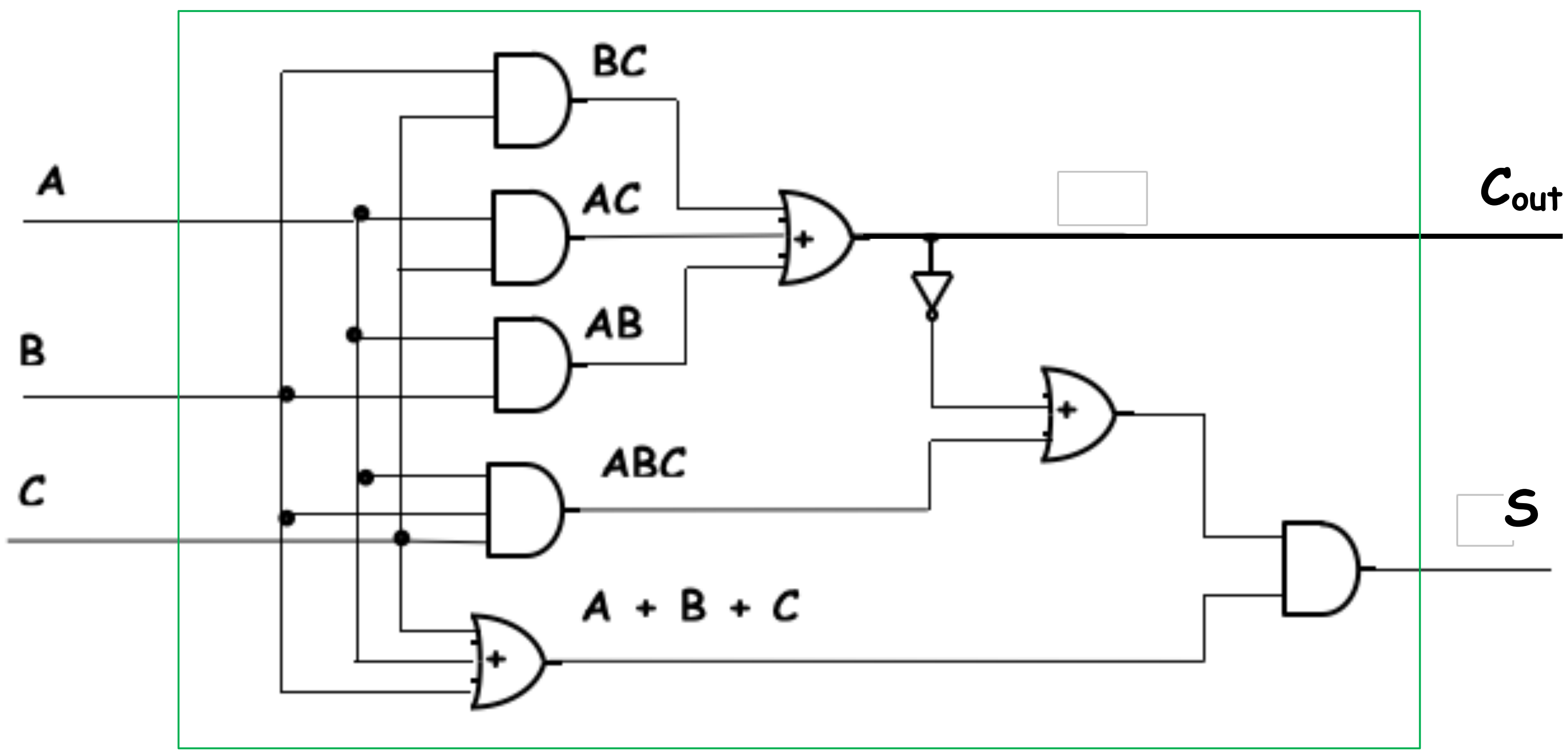
Prove we get:

**FA** $S = A'B'C + A'BC' + AB'C' + ABC$
$C = AB + AC + BC$

**HW 18.1 - assigned**

Continue this proof. Show $S = S_2$ and $C_{out} = C_1 + C_2$

## IBM FA:



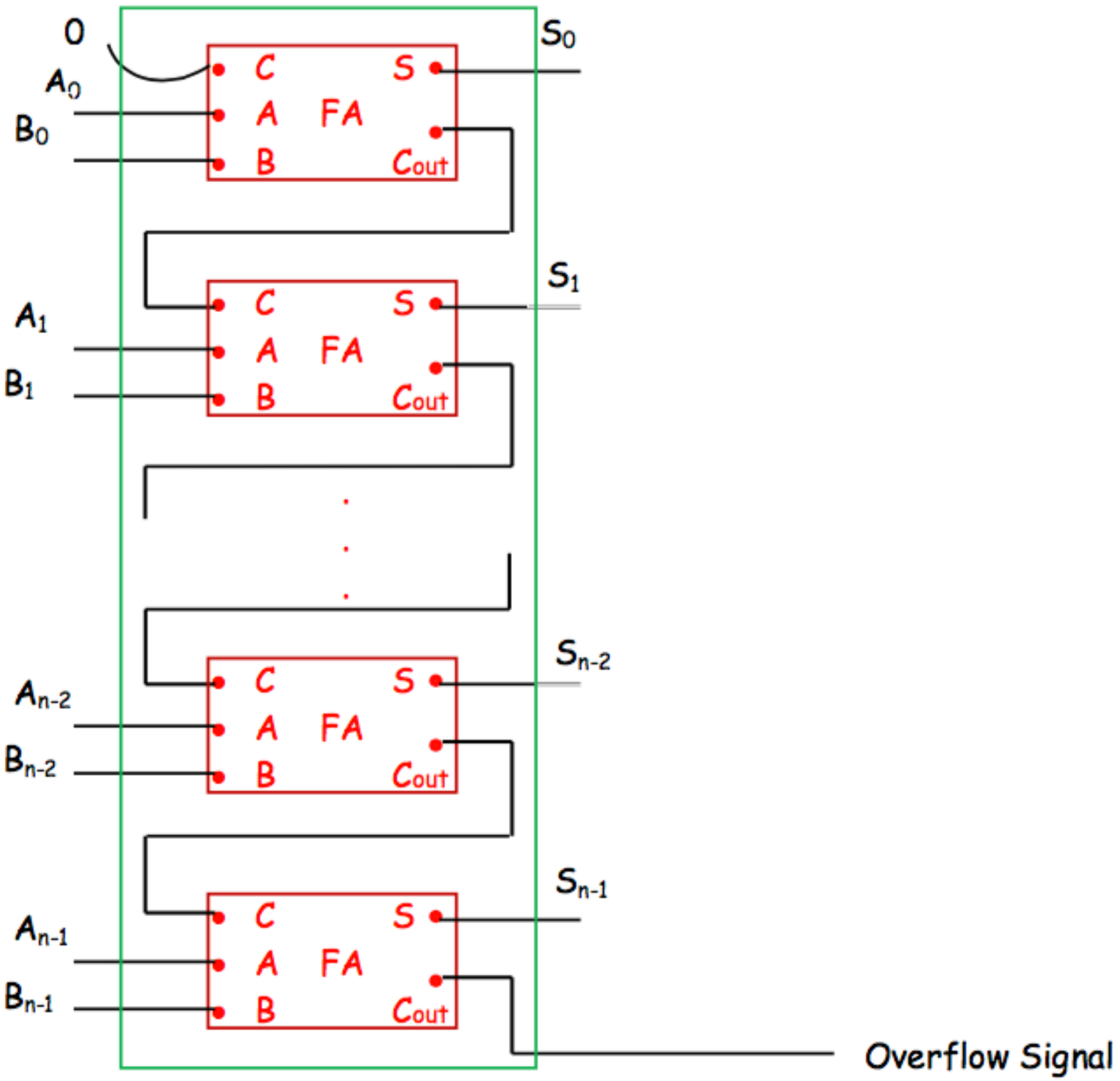

HW 18.2 - assigned

Prove it is indeed a FA, i.e. it creates the functions S, $C_{out}$ of a FA.

## Adding multiple digit numbers

Suppose we have two n-digit binary numbers: $A = A_{n-1}\, A_{n-2} \ldots A_1\, A_0$ **and** $B = B_{n-1}\, B_{n-2} \ldots B_1\, B_0$

We obtain their sum $S = S_{n-1}\, S_{n-2} \ldots S_1\, S_0$ using binary FAs, by adding them bit by bit starting with the lsd's:



# Half-Subtractor and Full-Subtractor

## HS:

Like HA, it has
2 inputs and 2 outputs.

$$\begin{array}{r} \beta_{out} \leftarrow \quad \\ A - \\ \underline{B \quad} \\ D \quad \end{array}$$

| A | B | D | $\beta$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |

$D = A'B + AB'$

$\beta = A'B$

## FS:

Like FA, it has
3 inputs and 2 outputs.

$$\begin{array}{r} \beta_{out} \leftarrow \quad \leftarrow \beta_{in} \\ A - \\ \underline{B \quad} \\ D \quad \end{array}$$

| A | B | $\beta_{in}$ | D | $\beta_{out}$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 |

**HW 18.3 - assigned**

Finish and minimize D, $\beta_{out}$ for FS.

**HW 18.4 - assigned**

Construct a FS using only HS's and one other gate.

# COMPARATOR

We compare two 3-bit binary numbers: $A = A_2 A_1 A_0$

$B = B_2 B_1 B_0$

We define the functions:

$$\boxed{f_= = 1 \longleftrightarrow A = B}$$

$$f_= = (\overbrace{A_2B_2}^{A_2=1\ \&\ B_2=1} + \underbrace{A'_2B'_2}_{A_2=0\ \&\ B_2=0})\,(A_1B_1 + A'_1B'_1)\,(A_0B_0 + A'_0B'_0)$$

$$\underbrace{(A_2B_2 + A'_2B'_2)}_{A_2 = B_2}\,\underbrace{(A_1B_1 + A'_1B'_1)}_{A_1 = B_1}\,\underbrace{(A_0B_0 + A'_0B'_0)}_{A_0 = B_0}$$

$$\boxed{f_< = 1 \longleftrightarrow A < B}$$

$$f_< = \overbrace{A'_2B_2}^{A_2=0\ \&\ B_2=1} + (A_2B_2 + A'_2B'_2)(A'_1B_1 + (A_1B_1 + A'_1B'_1)\ A'_0B_0)$$

$$f_< = \underbrace{A'_2B_2}_{A_2 < B_2} + \underbrace{(A_2B_2 + A'_2B'_2)}_{A_2 = B_2}(\underbrace{A'_1B_1}_{A_1 < B_1} + \underbrace{(A_1B_1 + A'_1B'_1)}_{A_1 = B_1}\ \underbrace{A'_0B_0}_{A_0 < B_0})$$

**HW 18.5 - assigned**

Express the function: $\boxed{f_> = 1 \longleftrightarrow A > B}$