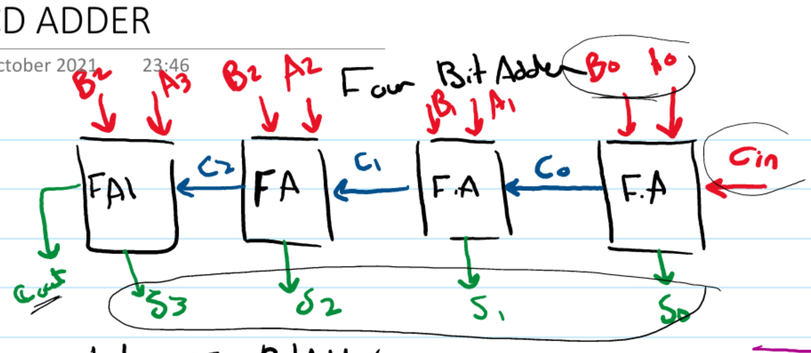
BCD ADDER

13 October 2021 23:46




```
entity FourBitAdder
    a,b : in  STD_LOGIC_Vector (3 downto 0);
    Cin : in  std_logic;
    S   : out STD_LOGIC_Vector (3 downto 0);
    Cout: out std_logic;
```

```
entity BCD_Adder is
port(
    a,b  → in  std logic vector (3 → 0)
    Cin  → in  std logic
    S    → out std_logic_Vector (3 → 0)
    Cout → out std logic
);
architecture struc of BCDAdder is
component FourBitAdder
    a,b  → in vector
    Cin  → in
    S    → out vector
    Cout → out
);
Signal Cout1 : std logic;
Signal Cout2 : std logic;
Signal Z : std logic vector (3 → 0)
Signal BCDin2 : std logic vector(3 → 0)
begin
FB1: FourBitAdder port map(a,b,Cin,Z,Cout1);
Cout2 <= Cout1 or (Z(3) and Z(2)) or (Z(3) and Z(1));
BCDin2(0) <= '0'
BCDin2(1) <= Cout2
BCDin2(2) <= Cout2
BCDin2(3) <= '0'
FB2: FourBitAdder port map(Z, BCDin2, '0', S, open);
Cout <= Cout2;
end arch;
```

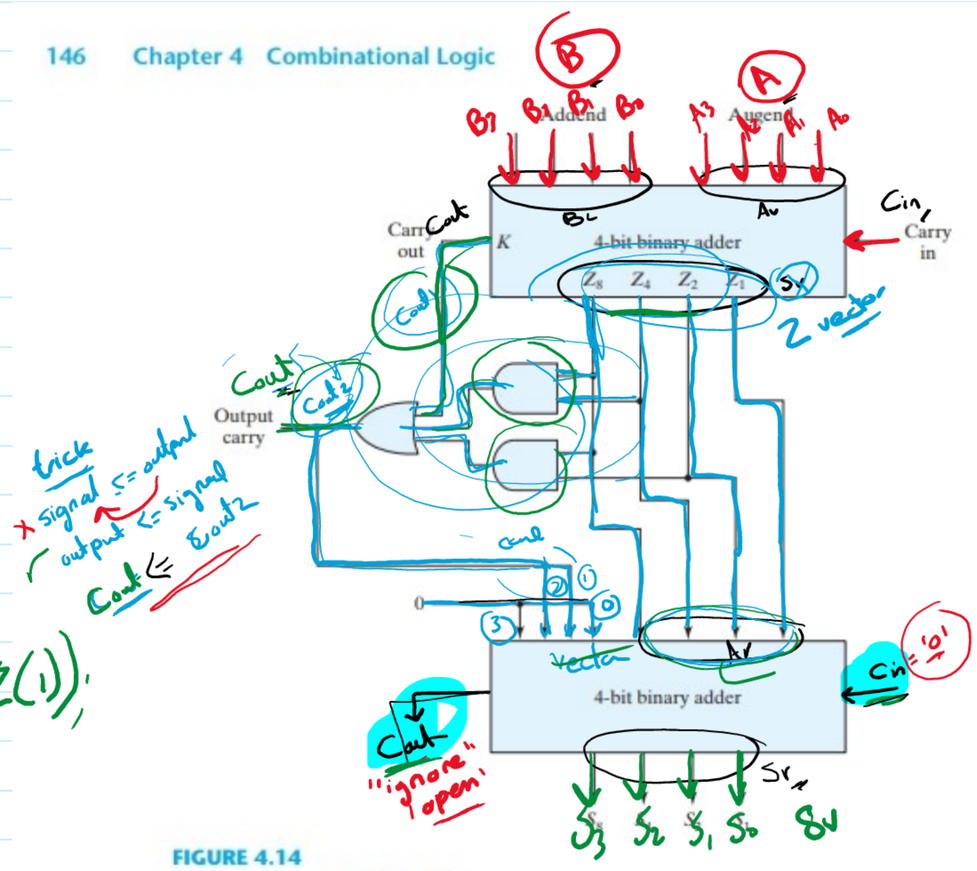



FIGURE 4.14
Block diagram of a BCD adder

Tricks
- ① Diagram
- cut signal cannot be inputs
  - ↳ create signal
    - output <= signal
- '0' no input
- 'open' ignore output