

# Previ DGD practica 2

Daniel Vilardell

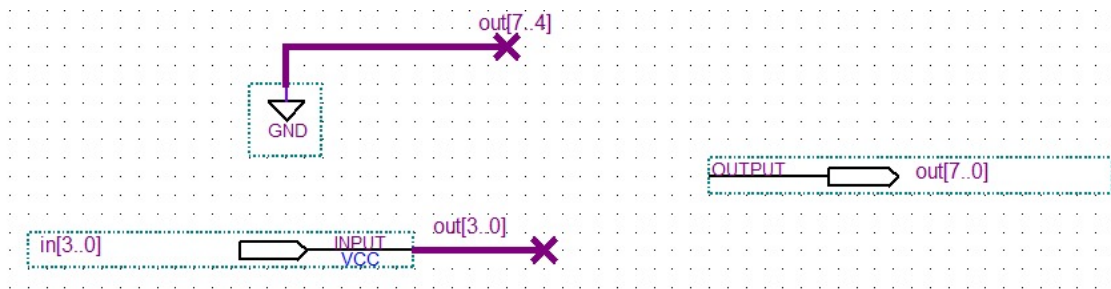
Igor Yuziv

**Question 1:** Per dissenyar el bloc combinacional AperB aprofitem part del disseny fet en la pràctica anterior, on justament es feia un multiplicador. Expliqueu com era aquest disseny i quines modificacions cal fer-li per poder utilitzar-lo a ppal.

Per tal de fer el bloc AperB, usarem el bloc multiplicador realitzat a la practica anterior, el que com que l'entrada es de dos nombres de 4 bits mentres que el component multiplicador se li entren dos entrades de 8 bits, haurem de fer un conversor de nombres en binari de 4 a 8 bits. La sortida ens la donara en binari, per a passarla a BCD usarem un component programat amb vhdl que considerarà tots els possibles nombres solucio i assignarà al resultat el que li toqui.

Primer de tot farem un breu resum de com funcionava el multiplicador. Aquest feia la multiplicació escolar, es a dir, colocava un nombre a dalt i un a baix i multiplicava xifra per xifra. A mesura que anava multiplicant extreia el nombre de menys pes de cada producte individual i el situava a la sortida. Els altres digits els sumava amb el següent. Això fins als ultims digits que tots eren destinats al output.

El component conversor de 4 a 8 bits serà el següent



El component que ens convertira la sortida del multiplicador de binari a BCD es el següent.

```

LIBRARY ieee; USE ieee.std_logic_1164.ALL;

ENTITY BIN_BCD_8B IS PORT (
    BIN : IN STD_LOGIC_VECTOR(7 downto 0);
    BCD : OUT STD_LOGIC_VECTOR(7 downto 0));
END BIN_BCD_8B;

ARCHITECTURE taula_veritat OF BIN_BCD_8B IS
    BEGIN --
        with BIN SELECT BCD <=
            "10011001" WHEN "00111000", -- 81
            "01110010" WHEN "01001000", -- 72
            "01100100" WHEN "01000000", -- 64
            "01010110" WHEN "00111000", -- 56
            "01010100" WHEN "00111000", -- 54
            "00101000" WHEN "00011100", -- 28
            "01001001" WHEN "00110001", -- 49
            "01001000" WHEN "00110000", -- 48
            "01000101" WHEN "00101101", -- 45
            "01000010" WHEN "00101010", -- 42
            "01000000" WHEN "00101000", -- 40
            "00110110" WHEN "00100100", -- 36
            "00110101" WHEN "00100011", -- 35
            "00110010" WHEN "00100000", -- 32
            "00110000" WHEN "00011110", -- 30
            "00101000" WHEN "00011100", -- 28
            "00100110" WHEN "00011011", -- 27
            "00100101" WHEN "00011001", -- 25
            "00100100" WHEN "00011000", -- 24
            "00100001" WHEN "00010101", -- 21
            "00100000" WHEN "00010100", -- 20
            "00011000" WHEN "00010010", -- 18
            "00010110" WHEN "00010000", -- 16
            "00010101" WHEN "00001111", -- 15
            "00010100" WHEN "00001110", -- 14
            "00010010" WHEN "00001100", -- 12
            "00010000" WHEN "00001010", -- 10
            "00001001" WHEN "00001001", -- 9
            "00001000" WHEN "00001000", -- 8

```

```

"00000111" WHEN "00000111", — 7
"00000110" WHEN "00000110", — 6
"00000101" WHEN "00000101", — 5
"00000100" WHEN "00000100", — 4
"00000011" WHEN "00000011", — 3
"00000010" WHEN "00000010", — 2
"00000001" WHEN "00000001", — 1
"00000000" WHEN "00000000", — 0
"—————" WHEN OTHERS;  END taula_veritat;

```

**Question 2:** Dissenyeu el mòdul combinacional sel en forma de logigrama amb un nombre mínim de portes lògiques estàndard.