

Exercici 4: ALU

NOTA: Per aquest exercici necessitareu GHDL i GTKWave.

NOTA: Aquest exercici serà objecte de lliurament per a la seva avaluació. La data de lliurament sortirà a la tasca corresponent al Moodle de l'assignatura.

En aquest exercici practicareu amb operacions aritmètiques sobre vectors de bits, tot experimentant els efectes de compartició de recursos.

Dissenyeu i escriviu el codi VHDL d'un circuit digital que implementi una ALU de 8 bits amb la declaració següent (trobareu templates a la carpeta de l'exercici):

```
library IEEE;
use IEEE.Std_logic_1164.all;
entity ALU is
    port ( A, B :      in Std_logic_vector(7 downto 0);
          Op  :      in Std_logic_vector(3 downto 0);
          F    :      out Std_logic_vector(7 downto 0);
          Cout :      out Std_logic;
          Equal :      out Std_logic);
end ALU;
```

Totes les sortides són combinacionals. No es tracta d'un circuit seqüencial. El funcionament de la ALU ha de ser el següent:

- La sortida `Equal` val '1' si i només si $A = B$, i '0' en qualsevol altre cas. Això és independent de l'operació que faci la ALU.
- A i B són de tipus `Std_logic_vector`, però totes les operacions s'avaluen utilitzant el tipus **signed**.
- L'entrada `Op` és de tipus `Std_logic_vector` de 4 bits, que configura 13 operacions diferents i 3 que no s'utilitzen (dummy). La Taula 1 mostra un recull de les operacions.
- Les operacions de desplaçament (shift) són lògiques, és a dir, que els bits que deixa *buits* una operació shift a dreta o esquerra, s'omplen amb '0'.
- Un 'shift left' desplaça el bit 7 a `Cout`.
- Un 'rotate left' desplaça el bit 7 a `Cout` i també al bit 0.
- Per totes les operacions tret de 'shift' i 'rotate' les entrades A i B s'interpreten com a nombres binaris de 8 bits de tipus **signed** (notació complement a 2), i el resultat és un nombre **signed** de 9 bits els 8 bits de menys pes s'assignen a `F`, i el de més pes a `Cout`, que és en realitat el bit de signe. Si `Cout` és '1' indica que el resultat de l'operació és negatiu.

[illegible]