

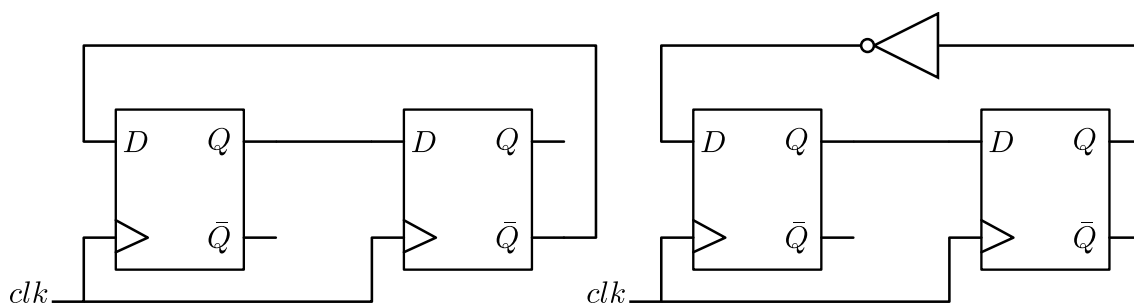
Introducció als Sistemes Digitals

Prova Final. 16 de gener de 2014

Temps per a la resolució: 3 hores. Publicació de resultats: 30 de gener.

1 Exercicis curts

- (4 points) When should we use pull-up and pull-down resistors? Give an example. Use as many (technical) words in English as you can.
- (10 punts) Implementeu la funció lògica $F_1(A, B, C, D) = ABC\bar{D} + AC + \bar{A}\bar{B}\bar{C} + ABC\bar{D}$. amb les següents restriccions (prèviament simplifiqueu la funció):
 - Useu un DEMUX amb el nombre de sortides que minimitzi la circuiteria extra.
 - Useu un MUX amb el nombre d'entrades que minimitzi la circuiteria extra.
 - Useu un 4:1 MUX amb els selectors A i C .
 - Useu un 2:1 MUX amb el selector C .
- (6 punts) Considereu els comptadors de Johnson de la figura.



- Per al circuit de l'esquerra. Feu el cronograma que apareix després de *resetejar* els dos flip-flop (considereu retards zero).
- Per al circuit de l'esquerra. Considerant un temps de propagació t_p de 14 ns, un temps de *setup* t_s de 6 ns i un temps de *hold* t_h de 3 ns, determineu la màxima freqüència del senyal clk .
- Per al circuit de la dreta. Repetiu l'apartat anterior considerant, a més, el retard de la porta NOT t_{NOT} de 5 ns.

2 Comptador a diferents velocitats

(10 punts) Dissenyau un comptador de 0 a 3 (en decimal) codificat en binari mitjançant dos bits (B_1B_0). Aquest comptador pot experimentar canvis en el flanc de pujada d'un *clock* d'1 Hz. El comptador es desborda, de forma que el nombre que segueix al 3 és el zero. Aquest comptador està controlat per dos polsadors P_1 i P_0 , de forma que quan $P_1P_0=00$ el comptador no avança, quan $P_1P_0=01$ el comptador fa increments de +1, quan $P_1P_0=10$ el comptador fa increments de -1, i quan $P_1P_0=11$ el comptador fa increments de +2.

3 Eviteu l'aparició d'un glitch

(10 punts) Considereu la següent descripció en VHDL d'un circuit combinacional.

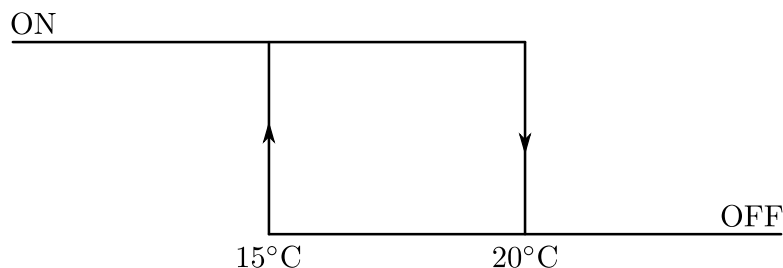
```
entity bloc is
port (a, b, c : in std_logic;
      y : out std_logic);
end entity;

architecture arch_bloc of bloc is
begin
y <= (not(a) and not(c)) or (not(a) and b) or (a and b and c);
end architecture;
```

- Escriviu la funció y que implementa aquest codi.
- Escriviu la seva funció simplificada y_s .
- Feu una representació del cronograma corresponent a la transició d' $abc = 011$ a $abc = 010$. Primer considereu la inexistència de retard i després un retard Δ per a totes les portes lògiques (AND, OR i NOT).
- Si observeu l'aparició d'un *glitch* afegiu redundància a la funció simplificada per tal d'implementar una nova funció y_r que eviti la possible aparició d'un *glitch*.

4 Un regulador de temperatura: ús del comparador amb histèresi

(10 punts) Us proposeu regular la temperatura T d'un habitacle. Disposeu de dos senyals provinents d'un sensor de temperatura. El senyal $A=0$ si $T < 15^\circ\text{C}$ i $A=1$ si $T > 15^\circ\text{C}$ i el senyal $B=0$ si $T < 20^\circ\text{C}$ i $B=1$ si $T > 20^\circ\text{C}$. També disposeu d'un sistema per escalfar l'habitable controlat per un senyal Y : quan $Y=0$ el sistema està ON, i quan $Y=1$ el sistema està OFF. El vostre objectiu és que per a $T < 15^\circ\text{C}$ el sistema estigui ON, per a $T > 20^\circ\text{C}$ el sistema estigui OFF i per a $15^\circ\text{C} < T < 20^\circ\text{C}$ mantingui l'estat anterior.



- Escriviu la taula de veritat de la sortida Y en funció de les entrades A i B .
- Escriviu la taula de veritat d'un *latch* SR: la sortida Q en funció de les entrades S i R .
- Implementeu aquest sistema usant un *latch* SR.
- Què passaria si el sensor funcionés incorrectament, de manera que el senyal A sempre valgués zero? Diferencieu la implementació del *latch* SR amb portes NOR i amb portes NAND.