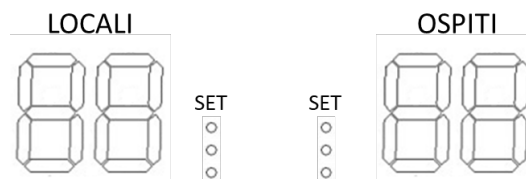


Prova d'esame di  
RETI LOGICHE (Prof. E. Torti)  
Lauree in Ing. Elettronica e Informatica DM 270  
24 settembre 2024

CANDIDATO:

n. matricola:

- 1) Si progetti un convertitore da codice BCD a codice in eccesso 7. Il circuito dovrà essere di costo minimo. Si realizzi poi il circuito relativo al secondo bit meno significativo con sole porte NAND e con sole porte NOR, indicando quale delle due soluzioni permette di ottenere il ritardo di propagazione minimo considerando lo stesso ritardo per le porte NAND e per le porte NOR.
- 2) Si progetti un sistema di allarme in grado di accendere un led L (L=0 led acceso, L=1 led spento), utilizzando il numero minimo di bit. Il led dovrà essere acceso quando all'interno della stanza monitorata la temperatura scende al di sotto di un set point impostato dall'utente. La temperatura viene rilevata da un sensore in grado di misurare un range tra  $[-5 \text{ e } 7] \text{ }^\circ\text{C}$  e risoluzione di  $1 \text{ }^\circ\text{C}$ . Il set point viene impostato dall'utente scegliendo valori all'interno del range  $[0;4]$  sempre con risoluzione pari ad  $1 \text{ }^\circ\text{C}$ . Si modifichi poi il progetto assumendo che il set point sia fisso a  $3 \text{ }^\circ\text{C}$ .
- 3) Si progettino un generatore ed un controllore di parità per un messaggio a 3 bit, discutendo come sia possibile variare il progetto in funzione del diverso tipo di parità considerato.
- 4) Si progetti il sistema di visualizzazione del punteggio da utilizzare in un campo di pallavolo. In particolare, si disegni lo schema a blocchi del circuito utilizzando contatori BCD, con ingresso di reset sincrono, per la visualizzazione del punteggio, connessi tramite opportuni decodificatori a dei display a 7 segmenti (i contatori BCD dovranno essere tutti uguali). La logica del punteggio dovrà essere gestita tramite comparatori, supponendo che il clock di alcuni contatori BCD sia connesso ad un pulsante che indica l'assegnazione del punto. Inoltre, dovranno essere presenti dei led, gestiti con un contatore in opportuna codifica binaria, che indichi il numero di set vinti. Dopo aver disegnato lo schema a blocchi si scrivano le equazioni dei contatori BCD, dei comparatori e del contatore dei set (non si indichino le equazioni dei decodificatori per i display).



- 5) Si progetti un circuito sommatore sequenziale, cioè è un circuito che, ad ogni ciclo di clock, riceve in ingresso la coppia di bit relativa agli operandi A e B da sommare, a partire dalla posizione meno significativa. Agli ingressi A e B del circuito viene fornita una sequenza binaria di lunghezza arbitraria, a partire dal bit meno significativo. Quando una coppia di bit si presenta agli ingressi A e B, il corrispondente bit della somma deve apparire immediatamente all'uscita Z. Per indicare che la sequenza è completa e che il circuito deve essere reinizializzato per ricevere un'altra sequenza, un altro ingresso Y diventa 1, mentre in tutti gli altri casi  $Y=0$ . Costruire il diagramma di stato, la tabella di stato e realizzare il circuito adottando la codifica Gray. Calcolare il costo degli ingressi assumendo che il costo di un singolo flip-flop sia pari a 14.

- 6) Si descrivano in dettaglio struttura e funzionamento dei due principali tipi di latch. Si descriva poi come modificare il latch SR per includere un ingresso di controllo C e come sia possibile ottenere un latch D, descrivendone anche il funzionamento.
- 7) Si consideri il seguente codice VHDL che descrive un flip-flop di tipo D. Commentare sinteticamente gli statement del codice (riportare sul foglio il numero della riga seguito dal relativo commento). Spiegare poi come dovrebbe essere modificato il codice per includere una porta di uscita che sia il complemento dell'uscita Q. Descrivere poi come andrebbe modificato il codice, cambiando il numero minimo di linee, per implementare un flip-flop di tipo JK. Descrivere poi come andrebbe modificato il codice, cambiando il numero minimo di linee, per implementare un flip-flop di tipo T.

```
1: library ieee;
2: use ieee.std_logic_1164.all;

3: entity dff is
4:     port(
5:         D, clk, reset : in std_logic;
6:         Q : out std_logic
7:     );

8: end dff;

9: architecture arch of dff is
10:     begin
11:         process(clk,reset)
12:             begin
13:                 if(reset='1') then
14:                     Q <= '0';
15:                 elsif(rising_edge(clk)) then
16:                     Q <= D;
17:                 end if;
18:             end process;
19: end arch;
```