

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

CANDIDATO:

n. matricola:

- 1) Si descrivano il funzionamento e la generica struttura di un multiplexer. Si indichino poi la tabella di verità e l'equazione booleana di un multiplexer 4-1. Si discutano poi le diverse modalità con cui potrebbe essere implementata la seguente funzione booleana tramite utilizzo di multiplexer, motivando quale risulta essere la più efficiente.

$$F = \Pi M(0, 1, 3, 4)$$

- 2) Si progetti il sistema di calcolo della differenza di temperatura tra l'interno di una stanza ed un valore di riferimento fisso a 20 °C, rappresentato su un numero opportuno di bit. Si assuma che le temperature all'interno della stanza vari in un range compreso tra 10 e 30 °C. La risoluzione del sistema deve essere di 1 °C. Calcolare il ritardo di propagazione totale del circuito assumendo che tutte le porte logiche utilizzate abbiano lo stesso ritardo di propagazione pari a 1 ns.
- 3) Si descriva la generica struttura di un blocco logico programmabile contenuto all'interno di un dispositivo FPGA, evidenziando nel dettaglio su quali elementi agiscono i bit di configurazione.
- 4) Un Si progetti un circuito in grado di riconoscere la sequenza 11011 all'interno di una sequenza arbitraria di bit. Il circuito dovrà essere dotato di un ingresso X e di un'uscita Z. L'uscita Z dovrà assumere il valore 1 tutte le volte che all'ingresso X sono già stati applicati i valori 1101 e l'ingresso corrente vale 1. In tutti gli altri casi l'uscita Z deve valere 0. Si preveda anche la presenza di un segnale di Reset asincrono che riporti il circuito nello stato iniziale. Si utilizzi un'opportuna codifica di costo minimo per assegnare codici agli stati e si utilizzino solo flip-flop di tipo D. Quanto vale il costo degli ingressi del circuito, considerando il costo di ogni singolo flip-flop pari a 14 (ignorare il costo delle porte NOT)? Esempio di sequenza da riconoscere:

X: 0 0 0 1 1 0 1 1 0 1 1 0
uscita Z: 0 0 0 0 0 0 0 1 0 0 1 0

- 5) Si progetto un contatore binario in complemento a 2 in grado di contare da -2 a 7, su un numero minimo di bit. Si considerino le combinazioni non utilizzate in fase di progetto come condizioni di non specificazione. Si realizzi il progetto utilizzando solo flip-flop di tipo D. Si indichi come andrebbe modificato il progetto per includere un segnale di ENABLE che permetta il normale conteggio quando ENABLE = 0 e blocchi il conteggio per ENABLE = 1; quando il circuito viene abilitato nuovamente dopo che il conteggio è stato bloccato, il conteggio deve ripartire dal valore al quale il conteggio era stato bloccato. Si calcoli, infine il costo totale degli ingressi delle equazioni originali, assumendo che il costo di un singolo flip-flop sia uguale a 14.
- 6) Si descrivano struttura e funzionamento di un contatore binario ad incremento e decremento. Sulle base delle considerazioni fatte, si progetti un contatore ad incremento e decremento a 4 bit.

- 7) Si consideri il seguente codice VHDL che descrive un sommatore/sottrattore a 5 bit. 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 un segnale di ENABLE attivo basso. Si indichi come andrebbe modificato il codice per rendere il circuito sincrono, aggiungendo un segnale di clock ed un segnale di RESET. Infine, includere un'uscita OVF che indichi il verificarsi della condizione di overflow.

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

4: entity addsub is
5:     port(
6:         A,B : in std_logic_vector(4 downto 0);
7:         sel : in std_logic;
8:         S   : out std_logic_vector(4 downto 0)
9:     );

10: end addsub;

11: architecture arch of addsub is
12:     signal res : signed(4 downto 0);

13:     begin
14:         with sel select
15:             res <= signed(A)+signed(B) when '0',
16:                 signed(A)+signed(not(B))+1 when '1',
17:                 "XXXXXX" when others;

18:         S <= std_logic_vector(res(3 downto 0));

19: end arch;
```