

Prova d'esame di
RETI LOGICHE (Prof. E. Torti)
Lauree in Ing. Elettronica e Informatica DM 270
4 settembre 2026

CANDIDATO:

n. matricola:

1. Si progetti un sistema di controllo di una serra. Il sistema ha i seguenti ingressi: 7 sensori di umidità U_x con $x=0,\dots,6$ ($U_x=1$ umidità corretta, $U_x=0$ umidità anomala), 4 sensori di temperatura T_y con $y=0,\dots,3$ ($T_y=1$ temperatura corretta, $T_y=0$ temperatura anomala), 3 sensori di acidità del terreno C_z con $z=0,\dots,2$ ($C_z=1$ acidità corretta, $C_z=0$ acidità anomala) ed un pulsante di abilitazione A ($A=1$ sistema abilitato, $A=0$ sistema non abilitato). Le uscite del sistema sono l'irrigatore I ($I=1$ irrigatore attivo, $I=0$ irrigatore non attivo) e la sirena S ($S=1$ sirena attiva, $S=0$ sirena spenta). Se il sistema non è abilitato, l'irrigatore e la sirena sono spenti. Se il sistema è abilitato ed umidità, temperatura ed acidità del terreno sono corrette, l'irrigatore è acceso e la sirena è spenta. In tutti gli altri casi, l'irrigatore è spento e la sirena è accesa. Si ricavino le equazioni minime del sistema e si calcoli il costo degli ingressi del sistema.
2. Si descriva la struttura di una look-up table (LUT) a 2 ingressi, indicandone l'utilità applicativa. Si spieghi poi come sia possibile utilizzare una LUT a 3 ingressi utilizzando solamente LUT a 2 ingressi e multiplexer 2-1. Infine, realizzare la seguente funzione booleana utilizzando solamente LUT a 2 ingressi e multiplexer 2-1.
$$F = AD + C\bar{D} + BC + \bar{A}\bar{B} + \bar{C}$$
3. Si descrivano la struttura ed il funzionamento di un circuito sommatore/sottrattore.
4. 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 progetti un contatore che esegua il conteggio da -3 a 4, rappresentando i numeri in complemento a 2 su un numero minimo di bit. Si esegua il progetto utilizzando solamente flip-flop D, considerando gli stati non utilizzati in fase di progetto come condizioni di non specificazione. Si disegni poi il diagramma di stato indicando anche cosa accadrebbe qualora si presentassero in ingresso combinazioni considerate come condizioni di non specificazione in fase di progetto. Si discuta come andrebbero gestiti questi casi in base al tipo di applicazione in cui verrà utilizzato il contatore.
6. Si descriva la struttura ed il funzionamento di un registro a caricamento parallelo con possibilità di scorrimento dati in entrambe le direzioni.

7. Si consideri il seguente codice VHDL che descrive un sommatore/sottrattore. 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 enable attiva bassa. Modificare poi il codice per rendere il circuito sincrono aggiungendo una porta per il segnale di clock e per il reset. Aggiungere infine una porta 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(3 downto 0);
7:         sel : in std_logic;
8:         S   : out std_logic_vector(3 downto 0)
9:     );

10: end addsub;

11: architecture arch of addsub is
12:     signal res : signed(3 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:                 "XXXX" when others;

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

19:     end arch;
```