

**Esercizio 0** (*punti 5*)

Rispondere alle seguenti domande:

1. Indicare quanti byte si possono memorizzare e quante funzioni combinatorie di quanti ingressi si possono realizzare con una ROM a 5 bit di indirizzo e 8 bit di dati. (*punti 1*).
2. Spiegare cosa si intende per overflow di un'operazione tra numeri binari a  $n$  bit e indicare come può essere identificato quando si esegue la somma tra numeri senza segno a  $n$  bit. (*punti 2*)
3. Descrivere **la struttura** del latch CD che si ottiene **usando un latch SR nella realizzazione**, indicando eventuali configurazioni vietate degli ingressi per un uso corretto del latch. (*punti 2*)

**Esercizio 1** (*punti 10*)

Una rete sequenziale asincrona riceve due segnali in ingresso  $s$  e  $t$  che non cambiano mai valore contemporaneamente, e non assumono mai contemporaneamente il valore "1". La rete genera un segnale di uscita  $q$ . Dopo un fronte di discesa di  $t$ ,  $q$  assume valore "0". Per mantenere  $q$  a "0",  $s$  deve assumere valore "1" esattamente una volta mentre  $t$  vale "0". Se questo non avviene, l'uscita deve segnalarlo assumendo valore "1" il prima possibile. All'avvio del sistema, la rete assume di aver appena visto un fronte di discesa di  $t$ .

1. Individuare **il grafo degli stati** utilizzando il modello di **Moore** e **dare una descrizione sintetica** della storia degli ingressi memorizzata in ogni stato. *(punti 3)*

2. Riportare **la tabella di flusso** corrispondente al grafo degli stati individuato. *(punti 1)*

**Prova d'esame di Reti Logiche T – 24/06/2019**

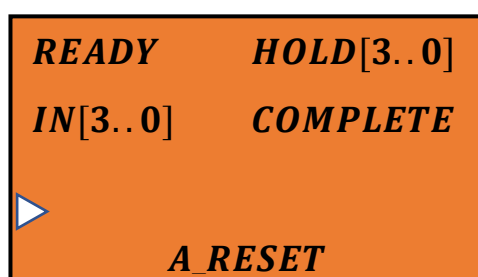
**COGNOME:**..... **NOME:** ..... **MATRICOLA:**.....

3. Individuare una **codifica degli stati** indicando il **grafo delle adiacenze** e la **tabella delle transizioni**, indicando e risolvendo **eventuali corse critiche**. (*punti 3*)

4. Individuare le **espressioni PS** di costo minimo della variabile di uscita e delle variabili di stato futuro, riportando le mappe di Karnaugh e i raggruppamenti rettangolari individuati (*punti 2*)

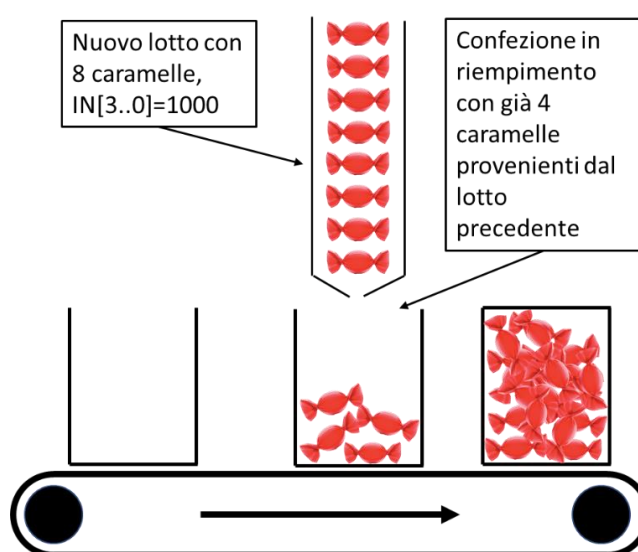
5. Disegnare lo schema logico della rete **comprensivo della rete di reset**. (punti 1)

### Esercizio 2 (punti 15)



Progettare in maniera diretta **minimizzando l'uso delle risorse** una rete sequenziale sincrona **pilotata da un clock a 8 Hz** che gestisce una macchina per il confezionamento di caramelle. La rete da progettare deve garantire che **ogni confezione contenga 32 caramelle**. Un nuovo lotto di caramelle è pronto in ogni periodo di

clock in cui l'ingresso sincrono **READY** assume valore "1". Contemporaneamente all'attivazione di **READY**, sul bus sincrono **IN[3..0]** viene comunicata alla rete la quantità di caramelle disponibili nel nuovo lotto **come numero intero senza segno**. Il valore presente sul bus **IN[3..0]** non è significativo quando **READY** vale "0". Alla ricezione di queste informazioni la rete deve comunicare alla macchina che si occupa del riempimento se il numero di caramelle del nuovo lotto consente di completare la confezione (considerate anche le caramelle inserite in precedenza): **se la quantità di caramelle disponibili permette di completare la confezione, la rete deve asserire il segnale di uscita COMPLETE e riportare come numero intero senza segno sul bus di uscita HOLD[3..0] quante caramelle del nuovo lotto NON devono essere usate per riempire la confezione attuale, ma inserite in quella successiva**. Per esempio, se nella confezione ci sono già 27 caramelle, e il nuovo lotto ne contiene 8, **COMPLETE** assumerà valore "1" e **HOLD[3..0]** valore 3. **Entrambe le uscite devono essere mantenute costanti per 2 secondi, a partire dal ciclo di clock in cui la rete ha ricevuto i dati in ingresso**. Si assuma che il segnale **READY** non possa assumere valore "1" durante questi 2 secondi. Se la rete non asserisce il



COGNOME:..... NOME: ..... MATRICOLA:.....

**segnale COMPLETE**, il valore sul bus HOLD[3..0] può essere qualsiasi e la macchina inserisce tutte le caramelle del nuovo lotto nella confezione attualmente in riempimento. Quando la macchina riceve il segnale COMPLETE=1, invece, inserisce nella confezione attuale (IN[3..0] – HOLD[3..0]) caramelle, fa procedere la confezione attuale alla stazione di lavorazione successiva, porta una nuova confezione vuota in posizione di riempimento e **inserisce le caramelle rimanenti del lotto nella nuova confezione**. La rete è dotata inoltre di un **segnale di ingresso asincrono A\_RESET**, che consente di inizializzarla all'accensione, quando in posizione di riempimento si trova sempre una confezione vuota.

