

Cognome e nome dello studente:

Matricola:

1. [9] **Esercizio obbligatorio.** Progettare e sintetizzare un controllore (una macchina a stati finiti, di Moore) che riconosca la parola “sos” all’interno di un testo. Determinare STG, STT, STT codificata e la struttura circuitale completa della macchina. Definire la complessità della parte combinatoria ed il suo cammino critico. Si può realizzare come macchina sincrona o asincrona? Perché? Cosa si intende per macchina sincrona e per macchina asincrona? Suggerimento: considerare come possibili input 3 insiemi diversi: “S”, “O”, “Tutti gli altri caratteri”.

2. [2] Quali tempi occorre considerare per definire il periodo del clock? Definire dei valori sensati per questi tempi e calcolare la frequenza del clock che si può ottenere per la macchina di cui sopra.

3. [2] Quale di queste frasi è corretta?

Uno shift register si può realizzare con latch di tipo D

Un flip-flop non si può utilizzare come elemento di memoria.

Un latch è più lento di un flip-flop, a parità di tecnologia.

Un latch non può essere sincronizzato.

Giustificare le risposte.

4. [3] Cos’è un segmento di memoria? In quali segmenti viene suddivisa la memoria gestita da un processore MIPS? Cosa si intende per codifica big-endian o little endian? Riguarda i dati o le istruzioni? Cosa rappresenta lo stack? Riempire il contenuto di un registro con i seguenti 4 byte contenuti in memoria, secondo la convenzione “big endian”, e scrivere quale numero viene letto.

Indirizzo	Contenuto del byte
0x4003	1
0x4002	13
0x4001	4
0x4000	57

5. [3] Costruire la porta di scrittura del register file e calcolare il cammino critico. Specificare in quali fasi del clock è possibile scrivere sul register file e perché.

6. [5] Scrivere il circuito del flip-flop di tipo SR, e scrivere la tabella di eccitazione e di transizione. Definire correttamente: bistabili, latch e flip-flop. Modificare il circuito del latch di tipo SR in modo tale che quando compare in ingresso la configurazione (S = 1, R = 1), il latch comunque cambi stato (=uscita).

7. [1] Indicare le modalità di indirizzamento dei dati previste dal MIPS.

8. [5] Tradurre **correttamente** in linguaggio macchina il seguente spezzone di codice Assembly:

```
Ciclo: 0x00001000 addi $t0, $t0, 1
       0x00001004 and $s2, $s3, $s4
       0x00001008 beq $t0, $s2 Salta:
       0x0000100C j Ciclo:
```

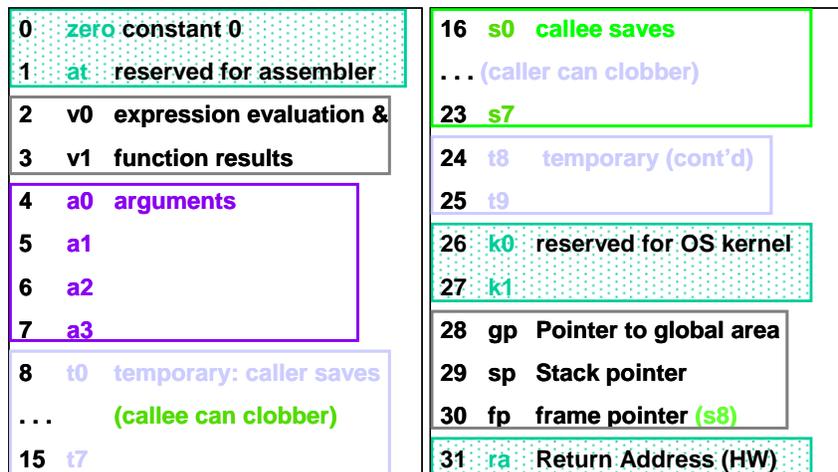
```
Salta: 0x01000000 sub $t5, $s6, $s2
```

9. [2] Tradurre correttamente in Assembly la seguente istruzione:

A[i] = a;

Definite voi un utilizzo appropriato dei registri.

10. [1] Definire cosa rappresenta la Jump Address Table e a cosa serve?



11. [2] Definire quali convenzioni di salvataggio dei registri vengono utilizzate nei processori MIPS. Cosa occorre fare per eseguire correttamente una chiamata a procedura ed il ritorno alla procedura chiamante? Quali sono gli elementi dell’architettura MIPS coinvolti?