Architetture dei Sistemi di Elaborazione   
06 febbraio 2023 – TURNO 3

Nome e Cognome\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Matricola\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Il codice compila senza errori: sì [ ] no [ ]

Ho provato il progetto in emulazione: sì [ ] no [ ]

Ho provato il progetto su board: sì [ ] no [ ]

L’ambiente di debug è stato utilizzato : sì [ ] no [ ]

Desidero ritirarmi [ ]

Leggere con attenzione:

* Occorre sviluppare un progetto ARM usando l’IDE KEIL µVision.
* Effettuare login su propria area al LABINF ed usare il software disponibile per editare, compilare e debuggare il codice.
* Utilizzare l’area desktop sul computer del LABINF per creare il vostro progetto.
* Utilizzare la scheda LANDTIGER o l’emulatore con tutte le non-idealità abilitate per debuggare il progetto
* Sono inibiti tutti gli accessi ad internet.
* Si possono utilizzare progetti esistenti, prelevati dalla propria chiavetta USB, ed è possibile consultare materiale cartaceo.
* Entro l’orario di consegna, occorre finalizzare il salvataggio di tutti i file (valido anche per la parte di modern architecture) e **copiarli nella propria area personale Z:/ all’interno della cartella che contiene le tracce**. Le consegne in ritardo (con file salvati oltre l’orario massimo di consegna) non vengono considerate valide e conducono in ogni caso all’insufficienza.
* In caso non sia possibile compilare con successo il progetto consegnato, la prova sarà considerata insufficiente. **Si richiede di predisporre l’ambiente di debug con le watch ed i breakpoint che permettano di seguire il flusso del programma**.

1. Si consideri un vettore **VETT** di **N** di valori interi con segno espressi su 32 bit. N è una costante dichiarata a tempo di compilazione e il vettore sarà dichiarato come **int VETT[N]**.
   * Per la fase di debug si consiglia di usare N pari a 4
   * Il programma deve in ogni caso funzionare con qualunque valore di N
   * Il contenuto iniziale di VETT deve essere 0 per ogni posizione.
2. In una prima fase, si devono acquisire gli elementi del vettore, derivandoli dal TIMER2 in collaborazione con il JOYSTICK:
   * Si programmi il Timer 2 per generare ciclicamente intervalli di 2 minuti e 50 secondi
   * Il Timer 2 NON scatena interruzioni alla fine del conteggio.
   * Ogni volta che il Joystick viene operato in direzione DOWN, il valore corrente del Timer 2 deve essere memorizzato nella prima posizione libera di VETT (inizialmente la prima posizione libera è la posizione 0).
   * VETT si comporta come un buffer circolare; se pieno (dopo N inserimenti), i valori catturati successivamente saranno scritti a partire dalla posizione 0, sovrascrivendo quelli ricevuti in precedenza



1. Durante la fase di acquizione dei valori, i led 10 e 11 mostrano in binario per quante volte è stato raggiunto il riempimento di VETT.
   * Ad ogni raggiungimento della posizione 0 dalla posizione N-1, il led 10 e 11 riportano un incremento e mostrano il valore in modo fisso.
   * A partire dalla quinta saturazione, i led sono resi entrambi lampeggianti con periodo di 500ms



1. In una seconda fase, alla pressione di INT0, viene invocata una funzione scritta in linguaggio ASSEMBLER in grado di analizzare il contenuto di VETT.
   * La funzione deve calcolare la differenza tra la media dei valori positivi e la media dei valori negativi registrati in VETT
   * Tale valore viene restituito al termine dell’esecuzione della funzione

Il prototipo della funzione è il seguente:

**int differenza\_media\_positivi\_negativi(int V[], unsigned int num, char\* over);**

* dove *num* indica la dimensione del vettore
* dove over viene utilizzato per restituire indirettamente indicazioni riguardanti le condizioni di overflow durante nel calcolo delle medie,
  + nel programma principale, over viene inizialmente impostato a 0
  + la funzione modifica over in caso si verifichi un overflow durante il calcolo delle medie, assegnandogli valore 0xFF

1. I led mostrano il risultato dell’esecuzione della funzione ASM, in base alle seguenti situazioni
   * Non si è verificato overflow e
     1. la differenza tra le medie è un numero positivo, allora tutti i led sono accesi fissi
     2. la differenza tra le medie è un numero negativo, allora solo il led 4 viene acceso fisso
   * Si è verificato overflow,
     1. indipendentemente dal valore restituito, tutti i led lampeggiano ad una frequenza di 3Hz.
2. Il valore visualizzato dai led viene mantenuto fino alla ulteriore pressione di INT0
   * Fino alla pressione di INT0, le funzioni di raccolta dati sono inibite
   * Alla pressione di INT0
     1. Il contenuto del vettore VETT viene “azzerato” (si riporta alla configurazione iniziale con valore 0 in tutti gli elementi)
     2. I LED sono tutti spenti
     3. Il processo ricomincia da 2) aspettando il valore da inserire alla posizione 0.

AGGIUNGI suffisso VS per controllo OVERFLOW (dopo S); SDIV e UDIV