Architetture dei Sistemi   
di Elaborazione

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 [ ]

Sistemi basati su ARM – 18 maggio 2022

Leggere con attenzione:

1. Occorre sviluppare un progetto ARM usando l’IDE KEIL µVision.
2. Effettuare login su propria area al LABINF ed usare il software disponibile per editare, compilare e debuggare il codice.
3. Utilizzare l’area desktop sul computer del LABINF per creare il vostro progetto.
4. Utilizzare la scheda LANDTIGER o l’emulatore con tutte le non-idealità abilitate per debuggare il progetto
5. Sono inibiti tutti gli accessi ad internet.
6. Si possono utilizzare progetti esistenti, prelevati dalla propria chiavetta USB, ed è possibile consultare materiale cartaceo.
7. 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.
8. 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 che permettono di seguire il flusso del programma.

Esercizio 1 (max 30 punti)

Si vuole realizzare un sistema in grado di effettuare misure temporali ed elaborarle.

Sviluppare le seguenti funzionalità per la scheda LANDTIGER ed il system-on-chip LPC1768.

1. Il sistema deve misurare in decimi di secondo il tempo di pressione del pulsante KEY1 e memorizzarlo in una variabile VAR di tipo Unsigned su 8 bit:
   * Ad ogni pressione del pulsante KEY1, il sistema deve calcolare e registrare in VAR il tempo intercorso dalla pressione al rilascio del pulsante.
   * Se il tasto viene premuto per un tempo più lungo del valore massimo memorizzabile su 8 bit, allora il valore scritto in VAR sarà il valore massimo rappresentabile.
2. Il valore in decimi di secondi registrato in VAR deve essere visualizzato tramite i LED; qualora il tempo sia superiore al massimo numero di decimi di secondo visualizzabili, allora tutti i LED risulteranno accesi e lampeggianti con una frequenza di 0,5 Hz
3. A valle di una misurazione, se viene premuto il pulsante KEY2,
   * Se tale pressione ha durata minore di 1 secondo, il valore corrente di VAR viene copiato nella prima posizione libera di un vettore chiamato VETT, istanziato di tipo opportuno e composto di 100 elementi
   * Altrimenti, il contenuto di VETT viene “svuotato” e lo stesso risulterà vuoto
4. Alla pressione del pulsante INT0 o in caso sia saturato il vettore VETT, deve essere lanciata la seguente funzione ASSEMBLER:

unsigned int totale\_pressioni\_con\_filtro(unsigned char VETT[], unsigned int   
 numero\_misure, unsigned char MAX, unsigned char MIN);

tale funzione riceve come parametri il vettore VETT, il numero di misure memorizzate e due valori chiamati MAX e MIN, definite come costanti a piacimento del programmatore e da utilizzare per filtrare i valori in VETT. La funzione restituisce la somma su 32 bit dei valori registrati che cadono tra MIN e MAX inclusi, mentre quelli fuori da questo intervallo devono essere esclusi.

1. Il risultato deve essere visualizzato tramite i led, i quali mostreranno il risultato a 32 bit suddiviso in 4 parti da 8 bit ciascuna. Iniziando dal byte più significativo, ogni parte deve essere visualizzata per 1,5 secondi. Durante il periodo di visualizzazione non sarà possibile acquisire altri valori.
2. Dopo 6 secondi, al termine della visualizzazione del risultato, il sistema riprende dal punto 1 con VAR e VETT vuoti così come inizialmente.