

TRACCIA – prova in presenza

Un sistema è composto da 3 unità A, B e C. B è collegato ad A mediante una periferica seriale, e a C mediante una periferica parallela. Il sistema opera come segue:

A invia fino ad un massimo di M messaggi di N byte a B. Per ogni messaggio ricevuto MSG_i , B verifica l'ultimo byte del messaggio $MSG_i(N-1)$:

- Se è diverso da 0, B continua con la ricezione;
- Se è uguale a 0, B interrompe la comunicazione con A e C.

Durante la ricezione dei messaggi (in qualsiasi momento), il sistema B può ricevere dei caratteri da C. In particolare, se riceve 2 caratteri (qualsiasi) successivi da C, B termina la ricezione del messaggio eventualmente in sospeso e poi interrompe la comunicazione con A e C.

Si sviluppi il progetto dell'unità B specificando:

- 1) *Architettura complessiva*: rappresentazione grafica schematica dell'architettura complessiva del sistema, in termini dei componenti di ciascuna unità (CPU, memoria, bus, dispositivi) e delle relative interconnessioni, in cui siano evidenziati i principali collegamenti e le linee di interruzione previste.
- 2) *Protocolli*: diagrammi temporali che rappresentino i principali protocolli di comunicazione utilizzati fra i dispositivi (ad es. i protocolli utilizzati per la scrittura e/o la lettura su/da periferica parallela).
- 3) *Mappa della memoria*: rappresentazione grafica schematica del contenuto della memoria RAM e ROM con riferimento alle aree dati e codice del programma implementato e al vettore delle eccezioni (solo per la specifica unità richiesta).
- 4) *Descrizione di alto livello del programma implementato*: descrizione, mediante diagramma a blocchi o pseudocodice, dei principali passi effettuati in ciascuno dei moduli software che compongono il programma (si richiede cioè un diagramma separato per il "main" e per ciascuna ISR prevista). In tale descrizione, lo studente deve specificare chiaramente le assunzioni fatte circa il comportamento delle periferiche, ad esempio legato alla gestione di possibili "conflitti" sull'accesso a dati globali e/o alla gestione di possibili "sovrapposizioni" di messaggi dovute alla diversa velocità di elaborazione dei dispositivi coinvolti.
- 5) *Implementazione*: codice Assembly Motorola 68000 per il sistema progettato. Gli studenti sono invitati a inserire commenti nel codice almeno nelle parti salienti (ad esempio, nella configurazione delle periferiche e nell'utilizzo di variabili globali) per favorire una migliore leggibilità e comprensione dell'elaborato.

Dopo aver sviluppato l'intero progetto, si illustri come cambierebbero l'architettura complessiva e la logica del driver se venissero inseriti un componente DMA e un componente PIC.

Nota: non è richiesta l'implementazione completa di un nuovo programma, ma lo studente dovrà indicare schematicamente le principali modifiche necessarie al codice assembly già prodotto (è preferibile a tale scopo indicare a parte gli stralci di codice da inserire ove necessario).

DOMANDE DI TEORIA

- 1) Lo studente descriva il flusso di esecuzione di una istruzione di tipo R e una di tipo J con riferimento alla pipeline del processore MIPS.
- 2) Illustrare le principali differenze esistenti tra processore MIPS e Motorola 68000 in relazione ad una operazione di swap fra due locazioni di memoria. Discutere le principali differenze fra processori RISC e processori RISC con riferimento ai modi di indirizzamento.
- 3) Descrivere sommariamente le tecniche per la gestione delle interruzioni precise evidenziando cosa accadrebbe se queste non fossero implementate.