

Prova intracorso di Computer System Design – prof. Mazzocca

12 maggio 2023

Un sistema è composto da 3 unità, A, B e C, tra loro collegate mediante due periferiche parallele che interconnettono A con B e A con C rispettivamente. Il sistema opera effettuando K iterazioni (con $K > 2$ a scelta dello studente), in ciascuna delle quali A deve ricevere globalmente 2 messaggi di N caratteri da B e 1 messaggio di N caratteri da C (con $N > 2$ a scelta dello studente). I messaggi da B e da C possono essere ricevuti in un ordine qualsiasi ma non deve essere mai possibile ricevere caratteri appartenenti a messaggi diversi intervallati tra di loro. In altre parole, detto msgB_i un generico messaggio completo ricevuto da B e msgC_j un generico messaggio completo ricevuto da C, in ogni iterazione si possono avere le seguenti situazioni:

<msgB_1 msgB_2 msgC_1> (s1)

<msgC_1 msgB_1 msgB_2> (s2)

<msgB_1 msgC_1 msgB_2> (s3)

Esempio: Un esempio con $K=3$ e $N=3$ di funzionamento del sistema è il seguente:

- Iter 1: msgB_1(1)msgB_1(2)msgB_1(3)msgC_1(1)msgC_1(2)msgC_1(3) msgB_2(1)msgB_2(2)msgB_2(3) (s3)
- Iter 2: msgC_1(1)msgC_1(2)msgC_1(3)msgB_1(1)msgB_1(2)msgB_1(3) msgB_2(1)msgB_2(2)msgB_2(3) (s2)
- Iter 3: :msgC_1(1)msgC_1(2)msgC_1(3)msgB_1(1)msgB_1(2)msgB_1(3) msgB_2(1)msgB_2(2)msgB_2(3) (s2)

Si progetti e implementi **l'unità A** 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 delle condizioni di funzionamento considerate e dei meccanismi usati per garantire lo svolgimento della logica prevista dall'esercizio* (es. è stato usato un flag che....); *descrizione di alto livello dei problemi ravvisati di conflitto* sui dati e/o di gestione di possibili "sovrapposizioni" di messaggi dovute alla diversa velocità di elaborazione dei dispositivi coinvolti) e *delle principali soluzioni scelte* (es. mutua esclusione con istruzione TAS, disattivazione selettiva interruzioni periferiche, ecc.).
- 5) *Descrizione di alto livello del programma implementato*: descrizione, mediante diagramma a blocchi o pseudocodice o automa, 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).
- 6) *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 venisse inserito un PIC. Opzionalmente, si discuta cosa accade inserendo un DMA (lo studente scelga la configurazione più opportuna) specificando le eventuali modifiche necessarie alla logica del programma.

Nota: per rispondere alla domanda su PIC/DMA 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).