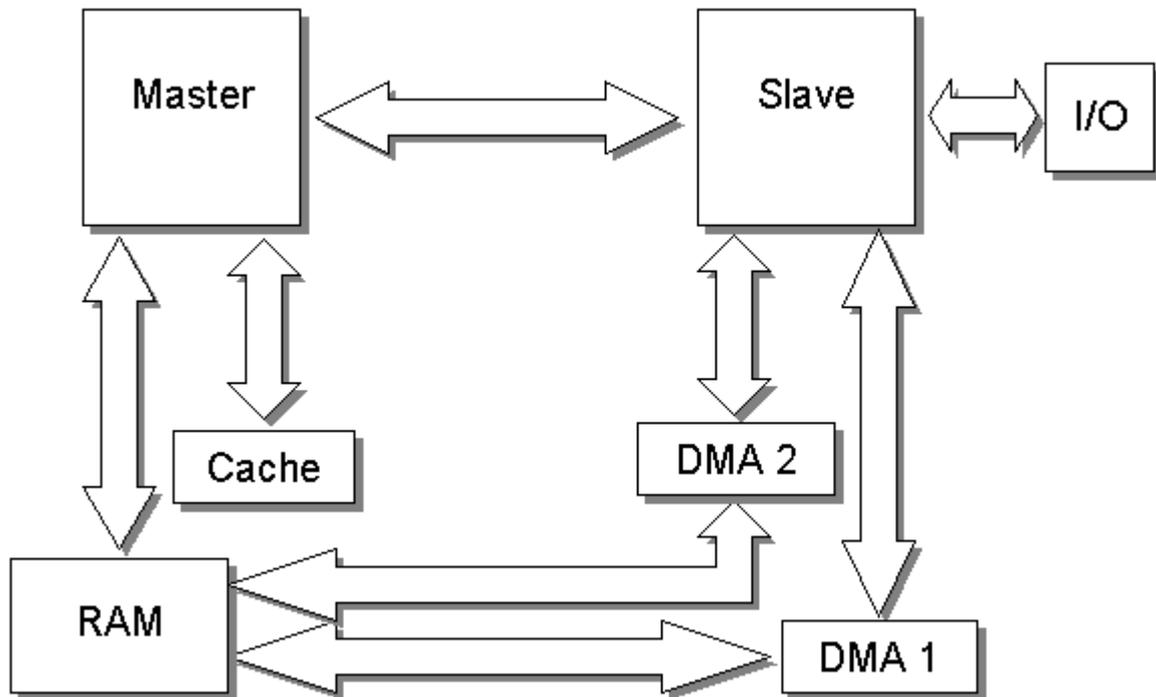


TRACCIA

Si devono progettare una coppia di microprocessori (M ed S). Il sistema complessivo è dotato di una RAM di 2^{24} celle da 8 bit, di una CACHE da 256KByte, di due canali DMA da 16 bit ciascuno, e di un canale di I/O, gestito con protocollo busy waiting, da 8 bit. Il processore M (*master*) è interfacciato direttamente con le memorie, mentre S (*slave*) è interfacciato con i dispositivi di I/O, secondo lo schema seguente:



S inoltre funge da coprocessore matematico.

I due canali DMA sono associati a due linee di interrupt, in grado di segnalare ad M la fine di un trasferimento DMA. Una terza linea di interrupt è usata da S per segnalare la fine dei trasferimenti da e verso il canale di I/O, mentre una quarta linea è usata per segnalare la fine di una operazione matematica.

In presenza di una interruzione, M salva il suo stato complessivo e salta ad una routine di servizio il cui indirizzo è presente a partire dalla locazione 0.

M è dotato del seguente linguaggio assembly:

CALL X

Salta alla subroutine presente alla locazione X.

RET

Ritorna alla subroutine chiamante. Deve essere possibile che una subroutine possa chiamare a sua volta un'altra subroutine.

FORWARD #C, X

Spedisce sul canale DMA #C la sequenza di n word presenti a partire dalla locazione X. n è dato dal valore del registro interno A.

RETRY #C, X

Recupera dal canale DMA #C, n word da memorizzare a partire dalla locazione X. Il valore di n è dato dal registro A.

RETI

Ritorno da una interruzione. Ripristina lo stato precedente de processore e salta al punto in cui si è verificata l'interruzione. Durante l'esecuzione della routine di servizio è possibile che si verifichino ulteriori interruzioni.

MASK C

L'operando a quattro bit C indica al processore quali interruzioni devono essere mascherate (disabilite).

WAIT C

M si sospende in attesa che si verifichi almeno uno degli interrupt indicati dalla maschera C.

READ X

Legge il dato X dal canale di I/O. M riprende l'esecuzione senza aspettare che l'operazione sia conclusa.

WRITE X

Scrive il dato X sul canale di I/O. M riprende l'esecuzione senza aspettare che l'operazione sia conclusa.

LOAD X

Carica nel registro A il valore della cella X.

STORE X

Memorizza nella locazione X il valore del registro A.

LOAD #X

Carica il valore assoluto #X nel registro A.

JZ X

Salta di X locazioni se il registro A vale 0. X è espresso in complemento a due.

ADD/SUB X

Somma/sottrae il contenuto della cella X ad A e pone il risultato in A.

MUL/DIV X

Chiede ad S di moltiplicare/dividere A per X. M riprende l'esecuzione senza attendere il risultato.

Si deve:

1. Implementare sia M che S;
2. Specificare il protocollo di scambio dati tra M ed S.
3. Implementare il controllo della cache ed il meccanismo di condivisione della RAM tra i canali DMA ed M.
4. Scrivere (in assembly M) una possibile routine di servizio, in grado di gestire ragionevolmente tutti e quattro i tipi di interruzione.