

Un processore z64 controlla la cassa di un supermercato. Esso è collegato a tre periferiche: LETTORE, che legge i codici a barre degli articoli; PULSANTE, che informa lo z64 che si richiede la stampa dello scontrino; STAMPANTE, che effettua tale stampa.

Ogni volta che un codice a barre viene letto, LETTORE invia un'interruzione allo z64. Il servizio associato all'interruzione di LETTORE consiste nelle seguenti operazioni:

- lo z64 legge dalla periferica il numero dell'articolo, a 16 bit; **REGISTRO**
- questo numero indica una posizione all'interno di un vettore di prezzi; ogni prezzo è una word e sono presenti 512 elementi nel prezzo.
- il prezzo viene copiato in coda alla lista dei prezzi degli articoli del cliente corrente. Questa lista è un vettore, la cui lunghezza massima è di 512 elementi.

Quando viene premuto il bottone della periferica PULSANTE, questa invia un'interruzione allo z64. Il servizio associato a questa interruzione deve:

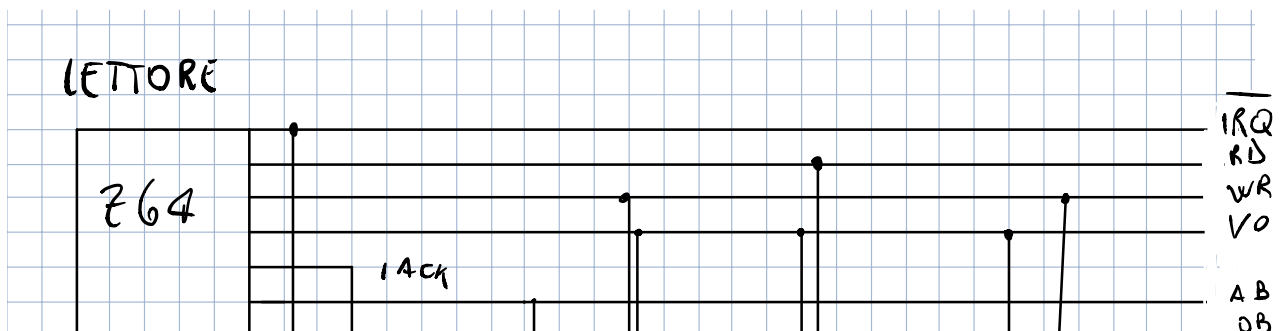
- inviare uno per volta (in busy waiting) i prezzi memorizzati fino a questo momento alla periferica STAMPANTE, che provvederà a stamparli su uno scontrino;
- inviare (in busy waiting) la somma dei prezzi a STAMPANTE, affinché la stampi;
- la lista dei prezzi deve essere svuotata, in modo che possa essere utilizzata per memorizzare i prezzi degli articoli del cliente successivo. In altre parole, quando si preme il pulsante, il prezzo del prossimo articolo letto (dopo la stampa dello scontrino) va nella posizione zero.

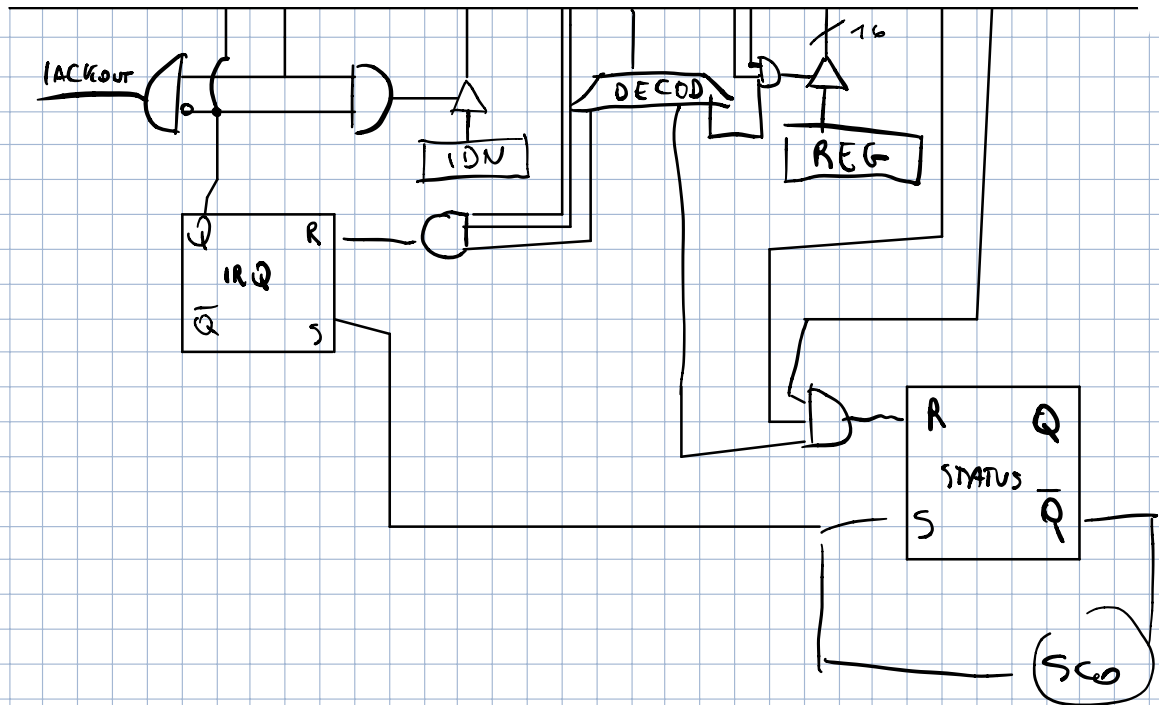
Tutte le routine di servizio devono essere non interrompibili e durante il trasferimento dei dati a STAMPANTE la periferica LETTORE non deve poter leggere altri codici a barre.

Progettare ed implementare:

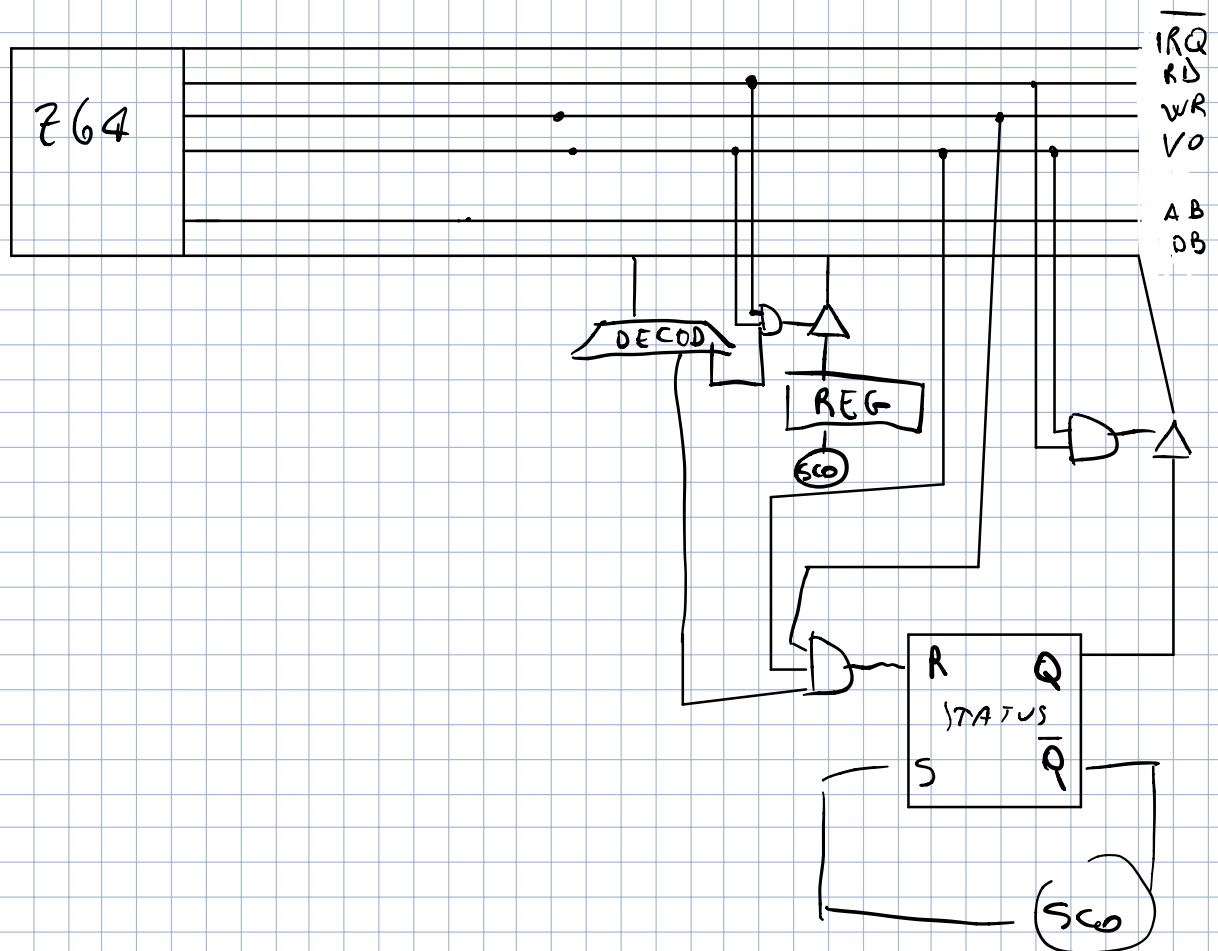
- le interfacce delle periferiche STAMPANTE e LETTORE
- i driver per la gestione delle interruzioni inviate da LETTORE e PULSANTE

SUGGERIMENTO: è possibile utilizzare un contatore memorizzato in una variabile globale per tenere traccia del numero di articoli processati da LETTORE.





STAMPANTE



.ORG 0X800

.DATA

.EQU LETTORE_STATUS, 0X0
.EQU LETTORE_IRQ, 0X1
.EQU LETTORE_REG, 0X2
.EQU STAMPANTE_REG, 0X3
.EQU STAMPANTE_STATUS, 0X4

PREZZI_CLIENTE: .FILL 512, 2, 0

PREZZARIO: .FILL 512, 2, ... (NON HO I PREZZI)

N_ELEM: .WORD 0

SOMMA: .WORD 0

.TEXT

MOVB \$1, %A2

MOVW \$LETORE_STATUS, %DX

OUTB %AL, %DX

MOVW \$STAMPANTE_STATUS, %DX

OUTB %AL, %DX

XORQ %R8W, %R8W

XORQ %R9W, %R9W

STI

HLT

.DRIVER 0 # DRIVER LETTORE

CLI