

MODALITÀ MISTA

- Un singolo dispositivo può operare sia in modalità busy waiting che in modalità asincrona
 - Ad esempio, per fornire funzionalità di natura e con latenza differente
- È sempre il processore a determinare la modalità operativa del dispositivo
- Un singolo dispositivo può supportare l'esecuzione di operazioni differenti
- In entrambi i casi è possibile programmare il dispositivo utilizzando un registro/flip-flop di opcode

COME FA IL PROCESSORE A DIRE AL DEVICE
COSA DEVE FARE? **OPCODE.**

↓
Deve essere bufferizzato.
Registro TAMPOE;

IL device:

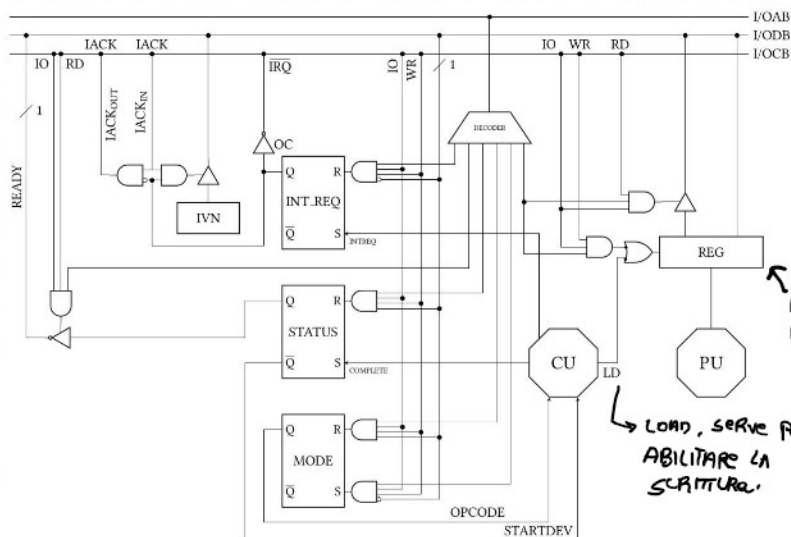
Quale attività può fare.

OPERARE SIA IN BUSY WAITING SIA CON LE INTERRUZIONI.

TRASFERIMENTO dati sia IN INPUT che OUTPUT!

DEVO DIRE AL MIO DISPOSITIVO: "PARTI, FAI QUALCHE COSA" 1:17:10 032

VEDIAMO LA SUA INTERFACCIA...



SE HO LA NECESSITÀ di DIRE AL MIO
DISPOSITIVO di PARTIRE e fare qualcosa,
ho bisogno per forza del mio FLIP-FLOP
di STATUS.

IL FLIP FLOP di STATUS LO ABBIAMO UTILIZZATO
COME "COMPARTITO" PER DIRE AL DISPOSITIVO
di PARTIRE.

SE RESETTO IL FLIP FLOP, IL VALORE di \bar{Q}
È IL SEGNALE START dev della mia
CU.

LA MACCHINA A STATI della CU, leggendo
questo segnale di START dev, capisce

che deve fare qualcosa.

Essendo un dispositivo che può voler lavorare in BUSY WAITING, PER IMPLEMENTARE IL BUSY WAITING
ABBIAMO la necessità di leggere il valore del flip flop, lo leggiamo con un'istruzione di
IN, NEL CASO IN CUI io voglio leggere il valore del flip flop, c'è un BUFFER THREE STATE che mi permette
di farlo.

HA QUESTO È ANCHE UN DISPOSITIVO CHE DEVE OPERARE TRAMITE IL MECCANISMO DELLE
INTERRUZIONI.

COME FA LA CPU A VEDERE LA BANDIERINA ALZATA? HA BISOGNO DEL FLIP FLOP DI INTERRUPT
REQUEST.

IL VALORE di QUESTO FLIP FLOP DETERMINA la richiesta di INTERRUZIONE da mandare al processore!

IL PROCESSORE RISPONDE CON IL SEGNALE di INTERRUPT ACKNOWLEDGE, che se ricevuto a causa
della richiesta di INTERRUZIONE, fa SCRIVERE L'IVN (INTERRUPT VECTOR NUMBER) SUL DATA BUS.
VICEVERSA SE ARRIVA un segnale di INTERRUPT ACKNOWLEDGE questo dispositivo NON ha mandato la richiesta di

della richiesta di INTERRUZIONE, fa scrivere l'IVN (INTERRUPT VECTOR NUMBER) sul data BUS. Viceversa se arriva un segnale di interrupt acknowledge questo dispositivo NON ha mandato la richiesta di INTERRUZIONE e il segnale di INTERRUPT acknowledge viene PROPAGATO IN DAISY CHAIN al dispositivo successivo.

CHI È CHE CANCELLA LA CAUSA DI RICHIESTA DI INTERRUZIONE? IL SOFTWARE. nel momento in cui scrive effettivamente sul flip flop;

CHI È CHE SETTA IL FLIP FLOP DI STATUS E IL FLIP FLOP DI INTERRUPT REQUEST?
L'UNITÀ di controllo (CU) del dispositivo;

COME FACIO A PROGRAMMARE IL MIO DISPOSITIVO PER DIRE "GUARDA che opero solo in BUS WAITING, oppure in una modalità basata solo sulle INTERRUZIONI"?

Per comunicare la modalità si manda un codice operativo, e questa modalità deve essere BULKERIZZATA, MI BASTA UN FLIP FLOP, che mi permette di specificare l'OPCODE verso la mia UNITÀ di CONTROLLO

E ci scrivo utilizzando un'istruzione di OUT!

Quindi utilizzando i segnali di controllo di: I/O e Write!

Mi prendo un bit dal data BUS e dico che se scrivo 1 lo RESETTO, se scrivo 0 lo SETTO! lo scelgo io!

IO, RD, WR, IACK SONO I SEGNALI di CONTROLLO che ho a disposizione!