

**Politecnico di Torino - III Facoltà di Ingegneria**  
**Esercitazioni del corso “Calcolatori Elettronici 2”**  
**Ing. Paolo Bernardi – email: [paolo.bernardi@polito.it](mailto:paolo.bernardi@polito.it)**

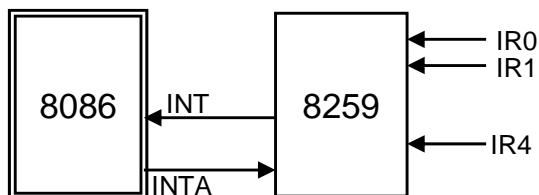
*Esercitazione 8 – Il controllore programmabile delle interruzioni 8259*

*Argomenti trattati:*

1. inizializzazione ed uso della periferica di gestione dell'interrupt 8259
2. inizializzazione della interrupt vector table
3. gestione di un sistema equipaggiato con la periferica di gestione dell'interrupt 8259.

**Esercizio 1:** Si inizializzi un dispositivo periferico 8259 mappato all'indirizzo 60H e la relativa Interrupt Vector Table in modo che:

- sia l'unico incluso nel sistema
- le interruzioni siano gestite sui fronti di salita
- sia adottata una strategia *non-specific end of interrupt*
- gestisca tre richieste di interruzione sui canali
  - o IR0
  - o IR1
  - o IR4
- sia gestito da un 8086



*Possibile svolgimento:*

```
; valori di inizializzazione
ICW1      EQU    00010011b      ; address 60H
ICW2      EQU    00100000b      ; address 61H
ICW4      EQU    00000001b      ; address 61H
OCW1_8259 EQU    11101100b      ; address 61H
OCW2_8259 EQU    00100000b      ; address 60H

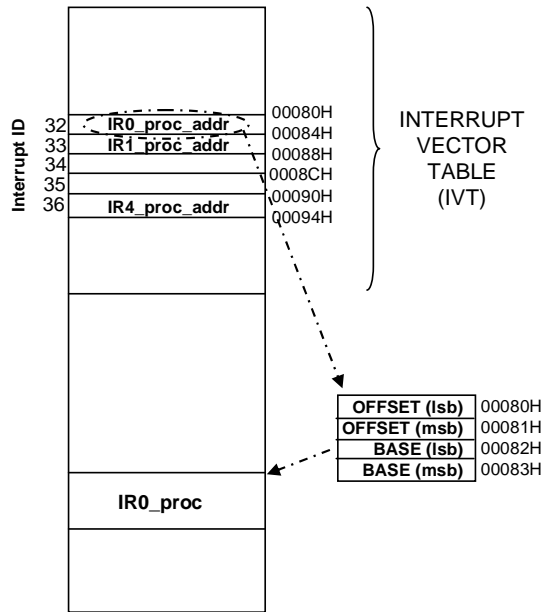
; funzione di inizializzazione dispositivo 8259
INIT_8259  PROC
          MOV    DX,    60H
          MOV    AL,    ICW1
          OUT    DX,    AL
          MOV    DX,    61H
          MOV    AL,    ICW2
          OUT    DX,    AL
          MOV    AL,    ICW4
          OUT    DX,    AL
          MOV    AL,    OCW1_8259
          MOV    DX,    61H
          OUT    DX,    AL
          RET
INIT_8259  ENDPROC

; funzione di inizializzazione della Interrupt Vector Table
INIT_IVT  PROC
          PUSH   AX
          PUSH   BX
```

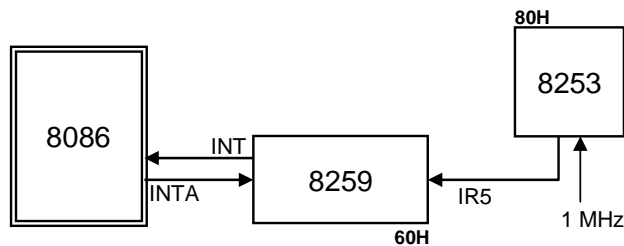
```

PUSH  DX
PUSH  DS
XOR   AX,  AX
MOV   DS,  AX
MOV   BX,  32          ; id interruzione IR0
SHL   BX,  2
MOV   AX,  offset IR0_PROC
MOV   DS:[BX],  AX
MOV   AX,  seg IR0_PROC
MOV   DS:[BX+2],  AX
MOV   BX,  33          ; id interruzione IR1
SHL   BX,  2
MOV   AX,  offset IR1_PROC
MOV   DS:[BX],  AX
MOV   AX,  seg IR1_PROC
MOV   DS:[BX+2],  AX
MOV   BX,  36          ; id interruzione IR4
SHL   BX,  2
MOV   AX,  offset IR4_PROC
MOV   DS:[BX],  AX
MOV   AX,  seg IR4_PROC
MOV   DS:[BX+2],  AX
POP   DS
POP   DX
POP   BX
POP   AX
RET
INIT_IVT  ENDP

```



**Esercizio 2:** Si programmi il sistema a microprocessore riportato in figura in modo che, dopo 10 microsecondi, il registro DI venga incrementato.



*Possibile svolgimento:*

```
; valori di inizializzazione
ICW1      EQU    00010011b
ICW2      EQU    00100000b
ICW4      EQU    00000001b
OCW1      EQU    11011111b
OCW2      EQU    00100000b
CW_8253   EQU    00010000b    ; modo 0 - 1 byte → val conteggio 10 dec - A hex
```

```
; programma main
.MODEL small
.STACK
.DATA
VETT      DB    50 dup (?)
.CODE
.STARTUP
CLI                          ; disabilita gli interrupt
CALL INIT_IVT
CALL INIT_8259
CALL INIT_8253
STI                          ; abilita nuovamente gli interrupt
MOV  DI,  0
ancora:   CMP  DI,  1
          JNE ancora
          .EXIT
```

; funzione di inizializzazione della Interrupt Vector Table

```
INIT_IVT  PROC
          PUSH  AX
          PUSH  BX
          PUSH  DX
          PUSH  DS
          XOR   AX,  AX
          MOV  DS,  AX
          MOV  BX,  37          ; id interruzione IR5
          SHL  BX,  2
          MOV  AX,  offset int_p
          MOV  DS:[BX],  AX
          MOV  AX,  seg int_p
          MOV  DS:[BX+2],  AX
          POP  DS
          POP  DX
          POP  BX
          POP  AX
          RET
INIT_IVT  ENDPROC
```

; funzione di inizializzazione dispositivo 8259

```
INIT_8259 PROC
          MOV  DX,  60H
          MOV  AL,  ICW1
          OUT  DX,  AL
          MOV  DX,  61H
          MOV  AL,  ICW2
          OUT  DX,  AL
          MOV  AL,  ICW4
          OUT  DX,  AL
          MOV  AL,  OCW1
          OUT  DX,  AL
```

```

RET
INIT_8259  ENDPROC

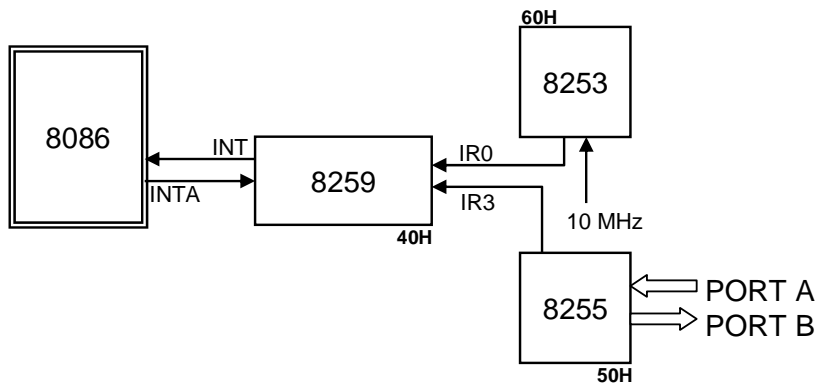
; funzione di inizializzazione dispositivo 8253
INIT_8253  PROC
MOV  DX,  83H
MOV  AL,  CW_8253
OUT  DX,  AL
MOV  DX,  80H
MOV  AL,  10
OUT  DX,  AL
RET
INIT_8253  ENDPROC

; funzione richiamata dall'interrupt sul canale IR5
int_p     PROC
INC  DI
MOV  DX,  60H
MOV  AL,  OCW2
OUT  DX,  AL
IRET
int_p     ENDP

END

```

**Esercizio 3:** Programmare il seguente sistema a microprocessore in modo che ogni 10 millisecondi la porta B del periferico 8255 venga scritta con l'ultimo dato letto dalla porta A sempre dell'8255.

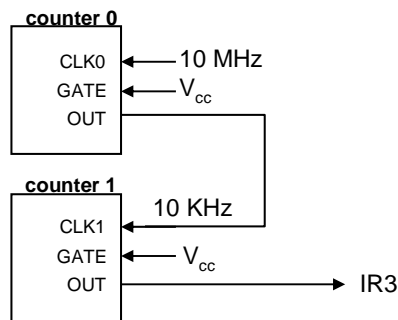


*Possibile svolgimento*

```

; configurazione 8253

```



```

; valori di inizializzazione

```

```

CW_8255      EQU    10110000b          ; PORT A modo 1 / PORT B modo 0
SPC4_8255    EQU    00001001b          ; per abilitare l'interrupt in lettura
CW0_8253     EQU    00110110b          ; val=1000=3E8H
CW1_8253     EQU    01010110B          ; val=100=64H
ICW1         EQU    00010011b
ICW2         EQU    00100000b
ICW4         EQU    00000001b
OCW1         EQU    11110110b
OCW2         EQU    00100000b

```

```

; programma main
.MODEL small
.STACK
.DATA
VETT         DB     50 dup (?)
.CODE
.STARTUP
CLI          ; disabilita gli interrupt
CALL INIT_IVT
CALL INIT_8259
CALL INIT_8253
CALL INIT_8255
STI          ; abilita nuovamente gli interrupt
ancora:     HALT
            JMP     ancora
            RET
            .EXIT

```

```

; funzione di inizializzazione della Interrupt Vector Table

```

```

INIT_IVT     PROC
            PUSH  AX
            PUSH  BX
            PUSH  DX
            PUSH  DS
            XOR   AX,  AX
            MOV   DS,  AX
            MOV   BX,  32          ; id interruzione IR0
            SHL   BX,  2
            MOV   AX,  offset scrivi
            MOV   DS:[BX],  AX
            MOV   AX,  seg scrivi
            MOV   DS:[BX+2],  AX
            MOV   BX,  35          ; id interruzione IR3
            SHL   BX,  2
            MOV   AX,  offset leggi
            MOV   DS:[BX],  AX
            MOV   AX,  seg leggi
            MOV   DS:[BX+2],  AX
            POP   DS
            POP   DX
            POP   BX
            POP   AX
            RET
INIT_IVT     ENDP

```

```

; funzione di inizializzazione 8259

```

```

INIT_8259    PROC
            MOV   DX,  40H
            MOV   AL,  ICW1
            OUT   DX,  AL
            MOV   DX,  41H
            MOV   AL,  ICW2

```

```

        OUT    DX, AL
        MOV    AL, ICW4
        OUT    DX, AL
        MOV    AL, OCW1
        OUT    DX, AL
        RET
INIT_8259  ENDP

; funzione di inizializzazione 8255
INIT_8255  PROC
        MOV    DX, 53H
        MOV    AL, CW_8255
        OUT    DX, AL
        MOV    AL, SPC4_8255
        OUT    DX, AL
        RET
INIT_8255  ENDP

; funzione di inizializzazione 8253
INIT_8253  PROC
        MOV    DX, 63H
        MOV    AL, CW1_8253
        OUT    DX, AL
        MOV    DX, 61H
        MOV    AL, 64H
        OUT    DX, AL
        MOV    DX, 63H
        MOV    AL, CW0_8253
        OUT    DX, AL
        MOV    DX, 60H
        MOV    AL, E8H
        OUT    DX, AL
        MOV    AL, 03H
        OUT    DX, AL
        RET
INIT_8253  ENDP

scrivi    PROC
        MOV    AL, BL
        MOV    DX, 51H
        OUT    DX, AL
        MOV    DX, 40H
        MOV    AL, OCW2
        OUT    DX, AL
        IRET
scrivi    ENDP

leggi     PROC
        STI
        IN     AL, 50H
        MOV    BL, AL
        MOV    DX, 40H
        MOV    AL, OCW2
        OUT    DX, AL
        CLI
        IRET
leggi     ENDP

        END

```