

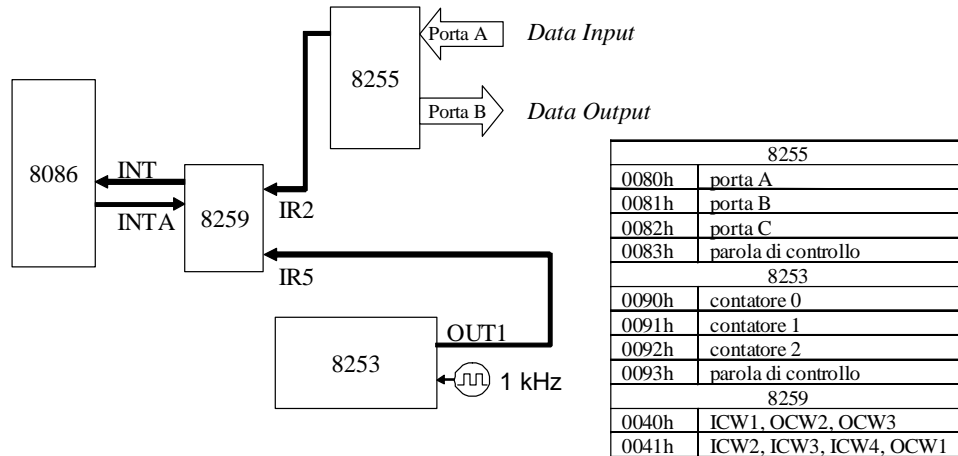
**Politecnico di Torino - III Facoltà di Ingegneria**  
**Esercitazioni del corso “Calcolatori Elettronici 2”**  
**Ing. Paolo Bernardi – email: [paolo.bernardi@polito.it](mailto:paolo.bernardi@polito.it)**

*Esercitazione 10 – Gestione di sistemi a microprocessore completi*

Argomenti trattati:

1. gestione di sistemi a microprocessore completi.

**Esercizio 1:** Si faccia riferimento al sistema di acquisizione dati mostrato nella figura seguente:



I dati ricevuti in input sulla porta A del dispositivo 8255 sono letti dalla CPU attraverso una richiesta di interruzione inviata alla linea IR2 di un dispositivo 8259. Il processore ogni 2 secondi (in corrispondenza di una richiesta di interruzione proveniente sulla linea IR5) deve scrivere sulla porta B dell'8255 il valore medio dei dati letti dalla porta A negli ultimi 2 secondi. Qualora non arrivi nessun dato, il processore non deve effettuare la trasmissione.

Si programmi il sistema a microprocessore in modo che soddisfi le specifiche.

*Possibile svolgimento:*

```

; valori di inizializzazione
ICW1_8259 EQU 00010011b
ICW2_8259 EQU 00100000b
ICW4_8259 EQU 00000001b
OCW1_8259 EQU 11011011b
OCW2_8259 EQU 00100000b

CW1_8253 EQU 01110110b

CW_8255 EQU 10110000b
SPC4 EQU 00001001b

.MODEL small
IVT SEGMENT AT 0
IVT ENDS
.STACK
.DATA
VAR DW 0
.CODE

```

```

; procedura di inizializzazione del dispositivo 8253
INIT_8253 PROC
    PUSH AX
    PUSH DX
    MOV DX, 93H
    MOV AL, CW1_8253
    OUT DX, AL
    MOV DX, 91H
    MOV AL, D0H ; ck = 1 KHz, ritardo = 2 sec
                ; → valore di inizializzazione 2000 = 7D0H
    OUT DX, AL
    MOV AL, 07H
    OUT DX, AL
    POP DX
    POP AX
    RET
INIT_8253 ENDP

```

```

; procedura di inizializzazione del dispositivo 8259
INIT_8259 PROC
    PUSH DX
    PUSH AX
    MOV DX, 40H
    MOV AL, ICW1_8259
    OUT DX, AL
    MOV DX, 41H
    MOV AL, ICW2_8259
    OUT DX, AL
    MOV AL, ICW4
    OUT DX, AL
    MOV AL, OCW1_8259
    MOV DX, 41H
    OUT DX, AL
    PUSH DX
    PUSH AX
    RET
INIT_8259 ENDP

```

```

; procedura di inizializzazione del dispositivo 8255
INIT_8255 PROC
    PUSH DX
    PUSH AX
    MOV DX, 83H
    MOV AL, CW_8255
    OUT DX, AL
    MOV AL, SPC4
    OUT DX, AL
    POP AX
    POP DX
    RET
INIT_8255 ENDP

```

```

; procedura 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, 34
    SHL BX, 2

```

```

MOV     AX,    offset IR2_P
MOV     DS:[BX],    AX
MOV     AX,    seg IR2_P
MOV     DS:[BX+2],    AX
MOV     BX,    37
SHL     BX,    2
MOV     AX,    offset IR4_P
MOV     DS:[BX],    AX
MOV     AX,    seg IR4_P
MOV     DS:[BX+2],    AX
POP     DS
POP     DX
POP     BX
POP     AX
RET
INIT_IVT    ENDP

```

```

; procedure di interrupt

```

```

IR2_P      PROC                                ; lettura da port A 8255
           PUSH AX
           PUSH DX
           INC DI
           MOV DX, 80H
           IN AL, DX
           CBW
           ADD [SI], AX
           MOV DX, 40H
           MOV AL, OCW2_8259
           OUT DX, AL
           POP DX
           POP AX
           IRET
IR2_P      ENDP

```

```

IR5_P      PROC                                ; scrittura su port B 8255 ogni 2 sec
           PUSH AX
           PUSH DX
           CMP DI, 0
           JE no_oper
           SUB SP, 2
           PUSH [SI]
           PUSH DI
           CALL MEDIA
           ADD SP, 4
           POP AX
           MOV DX, 81H
           OUT DX, AL
           XCHG AH, AL
           OUT DX, AL
           MOV DI, 0
           MOV [SI], 0
no_oper:   MOV DX, 40H
           MOV AL, OCW2_8259
           OUT DX, AL
           POP DX
           POP AX
           IRET
IR5_P      ENDP

```

```

; procedura di elaborazione dati

```

```

MEDIA     PROC
           MOV BP, SP

```

```

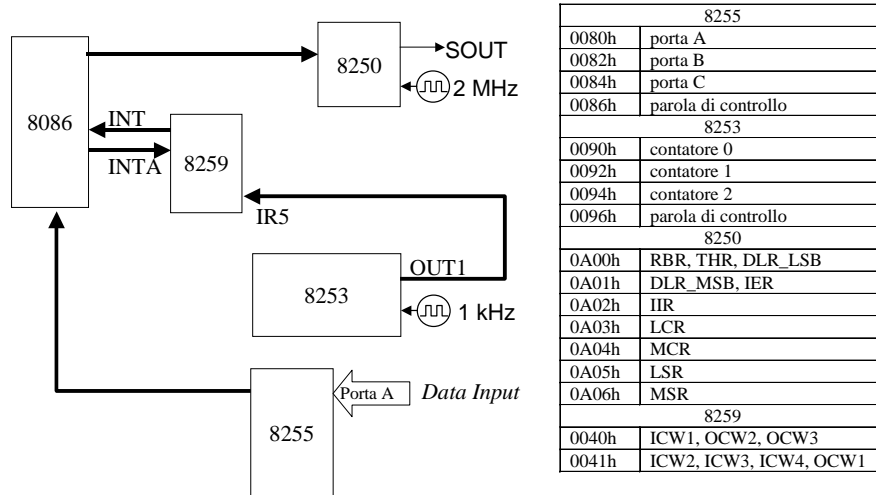
PUSH AX
PUSH BX
PUSH BP
MOV AX, [BP+2]
MOV BX, [BP+4]
CWD
DIV BX
MOV [BP+6], AX
POP BP
POP BX
POP AX
RET
MEDIA ENDP

; main program
.STARTUP
CLI ; disabilita tutti gli interrupt
CALL INIT_IVT
CALL INIT_8255
CALL INIT_8253
CALL INIT_8259
MOV DI, 0
LEA SI, VAR
STI ; abilita gli interrupt
ciclo:
NOP
JMP ciclo
.EXIT

END

```

**Esercizio 2:** Si faccia riferimento al sistema di acquisizione dati mostrato nella figura seguente:



Tenendo conto che il processore riceve ogni secondo una richiesta di interruzione proveniente dal dispositivo 8253, si realizzi il programma in grado di svolgere le seguenti operazioni:

- ogni 2 secondi il processore deve leggere dalla porta A del dispositivo 8255 un numero (rappresentato su un byte) compreso tra -20 e 100
- ogni minuto il processore deve scrivere tramite linea seriale il numero di valori letti che non sono compresi nel range specificato.

Si tenga conto che la trasmissione seriale asincrona avviene con i seguenti parametri:

- bit-rate: 9600 bps
- bit per carattere: 8
- parità: pari

- bit di stop: 1

*Possibile svolgimento:*

```
; valori di inizializzazione
```

```
CW1_8253 EQU 01110110b
```

```
ICW1_8259 EQU 00010011b
```

```
ICW2_8259 EQU 00100000b
```

```
ICW4_8259 EQU 00000001b
```

```
OCW1_8259 EQU 11011111b
```

```
OCW2_8259 EQU 00100000b
```

```
CW_8255 EQU 10010000b
```

```
LCR EQU 00011011b
```

```
IER EQU 00000000b
```

```
DLR EQU 000CH
```

```
.MODEL small
```

```
IVT SEGMENT AT 0
```

```
IVT ENDS
```

```
.STACK
```

```
.DATA
```

```
VAR DW 0
```

```
.CODE
```

```
; procedura di inizializzazione del dispositivo 8253
```

```
INIT_8253 PROC  
    PUSH AX  
    PUSH DX  
    MOV DX, 93H  
    MOV AL, CW1_8253  
    OUT DX, AL  
    MOV DX, 91H  
    MOV AL, D0H  
    OUT DX, AL  
    MOV AL, 07H  
    OUT DX, AL  
    POP DX  
    POP AX  
    RET
```

```
INIT_8253 ENDP
```

```
; procedura di inizializzazione del dispositivo 8255
```

```
INIT_8255 PROC  
    PUSH DX  
    PUSH AX  
    MOV DX, 83H  
    MOV AL, CW_8255  
    OUT DX, AL  
    POP AX  
    POP DX  
    RET
```

```
INIT_8255 ENDP
```

```
; procedura di inizializzazione del dispositivo 8259
```

```
INIT_8259 PROC  
    PUSH DX  
    PUSH AX
```



```

        PUSH AX
        PUSH DX
        CMP DI, 29
        JE scrivi
        MOV DX, 0080H
        IN AL, DX
        CALL RANGE
        INC DI
        JMP fine
scrivi: MOV DX, LSR
ciclo: IN AL, DX
        TEST AL, 00100000b
        JZ ciclo
        MOV DX, THR
        MOV AL, CL
        OUT DX, AL
        MOV DI, 0
        MOV CL, 0
fine:   MOV DX, 40H
        MOV AL, OCW2_8259
        OUT DX, AL
        POP DX
        POP AX
        IRET
IR5_P  ENDPROC

```

```

; procedura di elaborazione dati

```

```

RANGE  PROC
        PUSH AX
        PUSH BX
        CMP AL, 100
        JNL fine
        NEG AL
        CMP AL, 20
        JG fine
        INC CL
fine:   POP BX
        POP AX
        RET
RANGE  ENDPROC

```

```

; main program

```

```

.STARTUP
        CLI ; disabilita tutti gli interrupt
        CALL INIT_IVT
        CALL INIT_8255
        CALL INIT_8253
        CALL INIT_8259
        CALL INIT_8250
        MOV DI, 0
        MOV CL, 0
        STI ; abilita gli interrupt
ciclo: NOP
        JMP ciclo
        .EXIT
END

```