

---

# Informe Practica VI

---

Victor Tortolero & Oswaldo Capriles

Universidad de Carabobo  
Facultad de Ciencia y Tecnologia  
Arquitectura del Computador  
9 de diciembre de 2015

## 1. Las Interrupciones

Una interrupción es una señal que recibe el procesador, ante la cual debe detener cualquier proceso que este realizando y darle prioridad al proceso o tarea que debe realizar según la interrupción que haya ocurrido. Las interrupciones también puede decirse que son un evento, que es accionado por una señal de hardware o software.

Las interrupciones de hardware son usadas para indicar al sistema operativo que cierto dispositivo requiere atención. Y fueron hechas con la finalidad de evitar código innecesario que solo esperaba por una entrada o señal de algún hardware.

Cuando hablamos de las interrupciones por software, podemos hablar de las excepciones, que se usan para que el programa no reaccione de manera inesperada y sepa manejar los distintos errores que podrían causarse en tiempo de ejecución. Por ejemplo, en el caso de que el usuario ingrese una cadena y el programa espere un entero, en ese caso el programa no terminaría con un error, sino que haría algo al respecto, como notificarle al usuario y seguiría con la ejecución normal. Las interrupciones son comúnmente usadas para programas multitarea.

## **2. Programacion Concurrente**

Se habla de concurrencia cuando ocurren varios sucesos de manera contemporánea. En base a esto, la concurrencia en computación esta asociada a la ejecución de varios procesos que coexisten temporalmente.

Para definir correctamente la programación concurrente, es necesario diferenciar programa de proceso. Un programa es un conjunto de sentencias o instrucciones que se ejecutan secuencialmente, y un proceso es básicamente un programa en ejecución.

La concurrencia aparece cuando dos o mas procesos son contemporáneos. Un caso particular es el paralelismo o programación paralela. Los procesos pueden competir o colaborar entre si por los recursos del sistema. Por tanto, existen tareas de colaboración y sincronización

La programación concurrente se encarga del estudio de las nociones de ejecución concurrente, así como sus problemas de comunicación y sincronización.

## **3. Ejecución concurrente de programas y el temporizador**

En programación concurrente un temporizador es un objeto que puede notificar a un proceso si ha transcurrido un cierto intervalo de tiempo o se ha alcanzado una hora determinada, los temporizadores permiten crear periodos temporales que una vez finalizados pueden generar señales que se envíen a otros procesos o al sistema operativo, informando sobre la finalización del temporizador. De igual forma también se pueden utilizar como contadores de cuenta atrás. cada temporizador esta asociado a un reloj.

## 4. Código Fuente

```
1 ;Practica 6, Victor Tortolero, Oswaldo Capriles
2 DEFSEG INICIO1,ABSOLUTE
3 SEG INICIO1
4 JMP INICIO
5
6     ORG     1BH
7 SIT1: PUSH  ACC
8     CALL   boing
9     MOV    R7, #01H
10    POP     ACC
11    RETI
12
13 ORG     100H
14 ;Etiquetas de reloj
15 CERO EQU    30H
16 NUEVE EQU   39H
17 HOR1 EQU    20H
18 HOR2 EQU    21H
19 MIN EQU     24H
20 SEG EQU     27H
21 VAR EQU     50H
22 ;Etiquetas de carita
23 FACE EQU    02H
24 BLAN EQU    20H
25 FLAG EQU    F0
26
27 ;Subprograma de la Carita
28 boing: CJNE  R1, #3FH, swag
29     CPL     FLAG
30 swag: CJNE  R1, #38H, trans
31     CPL     FLAG
32 trans: JB    FLAG, izq
33 der: INC     R3
34     JMP     step
35 izq: DEC     R3
36 step: MOV    @R1, #BLAN
37     MOV    R1, 03H
38     MOV    @R1, #FACE
39     RET
40 ;Fin de Subprograma de la Carita
41
42 ;Proceso del reloj %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
43 RELOJ: MOV    R7, #00H
44     MOV    A, P0
45     MOV    R0, #SEG
46     CJNE  A, #'a', ELSE1
47     CALL  IRELOJ
48     RET
49 ELSE1: CJNE  A, #'A', ELSE2
50     CALL  IRELOJ
51     RET
```

```

52 ELSE2: CJNE A, #'d', ELSE3
53 CALL DRELOJ
54 RET
55 ELSE3: CJNE A, #'D', ELSE4
56 CALL DRELOJ
57 ELSE4: RET
58
59 IRELOJ: CJNE @R0, #'9', ELSE5 ;Revisando segundos
60 MOV @R0, #'0'
61 DEC R0
62 CJNE @R0, #'5', ELSE5
63 MOV @R0, #'0'
64 MOV R0, #MIN ;Revisando minutos
65 CJNE @R0, #'9', ELSE5
66 MOV @R0, #'0'
67 DEC R0
68 CJNE @R0, #'5', ELSE5
69 MOV @R0, #'0'
70 MOV R0, #HOR1 ;Revisando horas
71 CJNE @R0, #'2', ELSE6
72 INC R0
73 CJNE @R0, #'3', ELSE5
74 MOV @R0, #'0'
75 DEC R0
76 MOV @R0, #'0'
77 RET
78 ELSE5: INC @R0
79 RET
80 ELSE6: INC R0 ;Continuacion de la revision de horas
81 CJNE @R0, #'9', ELSE5
82 MOV @R0, #'0'
83 DEC R0
84 INC @R0
85 RET
86
87 DRELOJ: CJNE @R0, #'0', ELSE7 ;Revisando segundos
88 MOV @R0, #'9'
89 DEC R0
90 CJNE @R0, #'0', ELSE7
91 MOV @R0, #'5'
92 MOV R0, #MIN ;Revisando minutos
93 CJNE @R0, #'0', ELSE7
94 MOV @R0, #'9'
95 DEC R0
96 CJNE @R0, #'0', ELSE7
97 MOV @R0, #'5'
98 MOV R0, #HOR1 ;Revisando horas
99 CJNE @R0, #'0', ELSE8
100 INC R0
101 CJNE @R0, #'0', ELSE7
102 MOV @R0, #'3'
103 DEC R0
104 MOV @R0, #'2'

```

```

105         RET
106 ELSE7:   DEC    @R0
107         RET
108 ELSE8:   INC    R0          ;Continuacion de la revision de horas
109         CJNE   @R0, #'0', ELSE7
110         MOV    @R0, #'9'
111         DEC    R0
112         DEC    @R0
113         RET
114 ;Fin del Proceso del reloj %%%%%%%%%%%
115
116 ;Inicializando el Reloj en 00:00:00
117 INIT:    MOV    20H, #'0'
118         MOV    21H, #'0'
119         MOV    22H, #':'
120         MOV    23H, #'0'
121         MOV    24H, #'0'
122         MOV    25H, #':'
123         MOV    26H, #'0'
124         MOV    27H, #'0'
125         MOV    R1, #38H
126         MOV    @R1, #BLAN
127         INC    R1
128         CJNE   R1, #3FH, init2
129         MOV    @R1, #FACE
130         MOV    R3, 01H
131         RET
132 INICIO:   MOV    PSW, #00H
133         CALL   INIT
134         MOV    TMOD, #00100000B ;Poniendo Timer 0 y Timer 1 en modo 2
135         MOV    R7, #00H
136         SETB   EA          ;Activando Interrupciones
137         SETB   ET1         ;Activando interrupcion del Timer 1
138         SETB   TR1         ;Activando el Timer 1
139         SETB   PT1         ;Dandole prioridad al Timer 1
140
141 LOOP:    CJNE   R7, #01H, LOOP
142         CALL   RELOJ
143         JMP    LOOP
144
145 EXIT:    END

```

Listing 1: Codigo Practica 5

## Referencias

- [1] [https://es.wikipedia.org/wiki/Computaci%C3%B3n\\_concurrente](https://es.wikipedia.org/wiki/Computaci%C3%B3n_concurrente)
- [2] <http://www2.ulpgc.es/hege/almacen/download/20/20233/tema1.pdf>
- [3] [https://books.google.co.ve/books?id=8nTnCgAAQBAJ&pg=PT24&lpg=PT24&dq=Ejecuci%C3%B3n+concurrente+de+programas+y+el+temporizador&source=bl&ots=D27DHnY\\_fK&sig=t4YS5WZh7EwYU1s0eiRsCzVG3DE&hl=en&sa=X&redir\\_esc=y#v=onepage&q=Ejecuci%C3%B3n%20concurrente%20de%20programas%20y%20el%20temporizador&f=false](https://books.google.co.ve/books?id=8nTnCgAAQBAJ&pg=PT24&lpg=PT24&dq=Ejecuci%C3%B3n+concurrente+de+programas+y+el+temporizador&source=bl&ots=D27DHnY_fK&sig=t4YS5WZh7EwYU1s0eiRsCzVG3DE&hl=en&sa=X&redir_esc=y#v=onepage&q=Ejecuci%C3%B3n%20concurrente%20de%20programas%20y%20el%20temporizador&f=false)
- [4] <https://en.wikipedia.org/wiki/Interrupt>