

# Sistemas Digitales II

## Trabajo Final: Frecuencímetro

## Docentes

**Ing. Pedro Kolodka**

**Ing. Alejandro Münnemann**

## Alumnos

**Donna Gianetto**

**Daniel Ulloa**

# Implementar utilizando una placa de desarrollo

Programación en lenguaje  
Assembler.

Elección del método de  
medición.

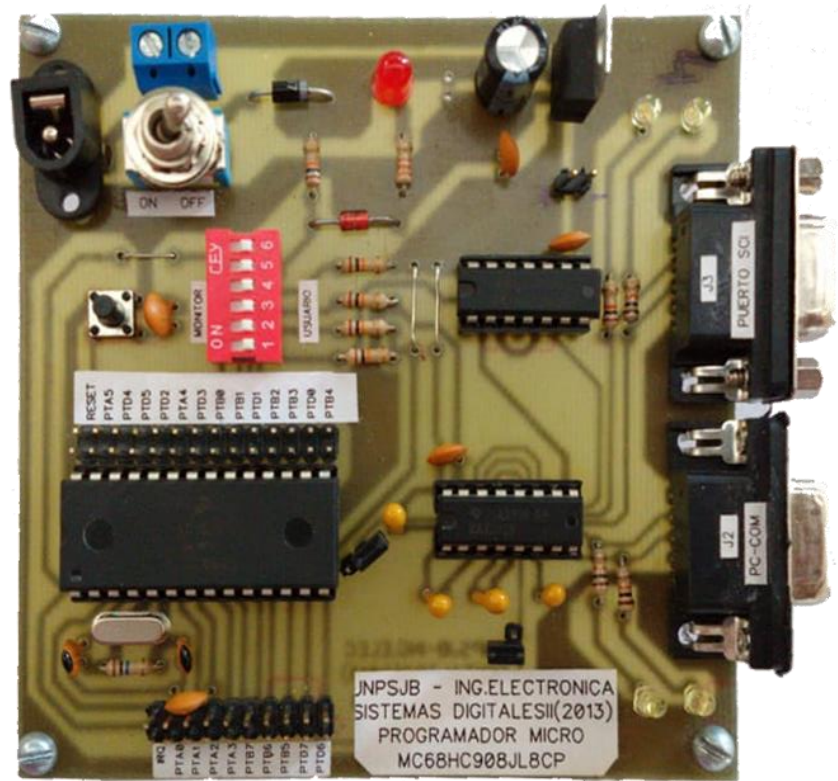
Configuración del hardware  
del microcontrolador.

Visualización en displays de  
7 segmentos.

---

# Placa de desarrollo

Permite programar el microcontrolador y conectar fácilmente circuitos externos. Se utiliza para el desarrollo de prototipos dónde se necesita cambiar constantemente el hardware.



# ¿Qué es un frecuencímetro?

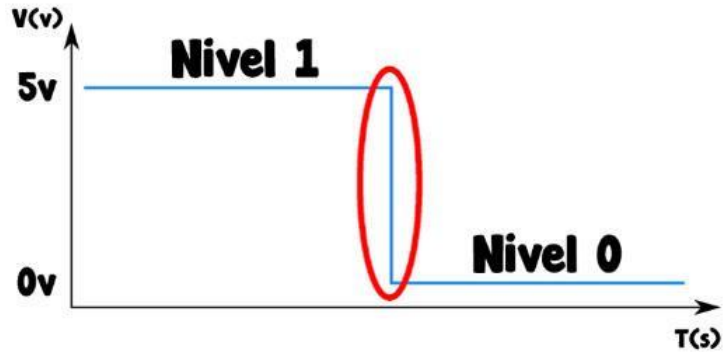
Es un instrumento de medición que se utiliza para conocer la frecuencia de una señal



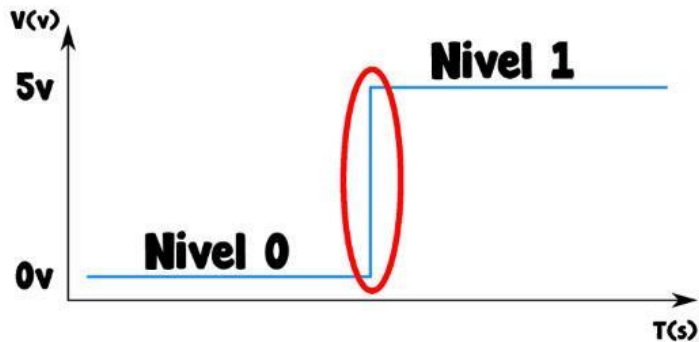
# Método de captura

Para medir la frecuencia de la señal se optó por contar los flancos ocurridos en una ventana de un segundo

# Flancos de una señal

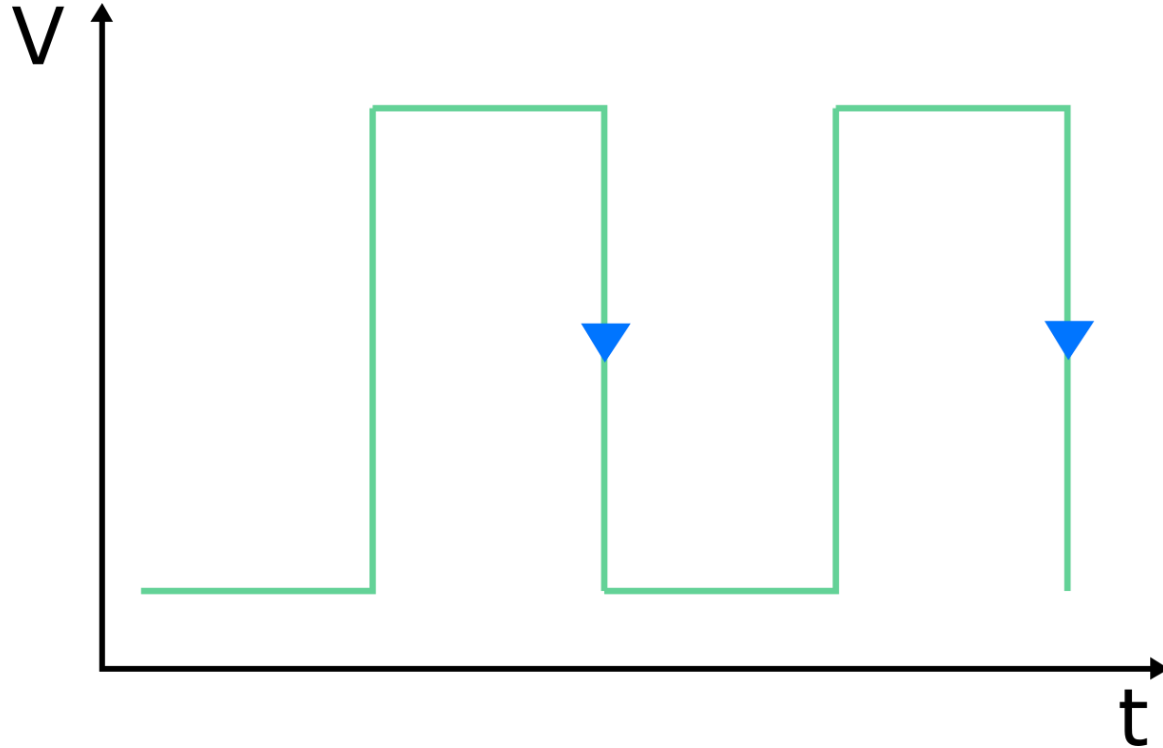


Descendente



Ascendente

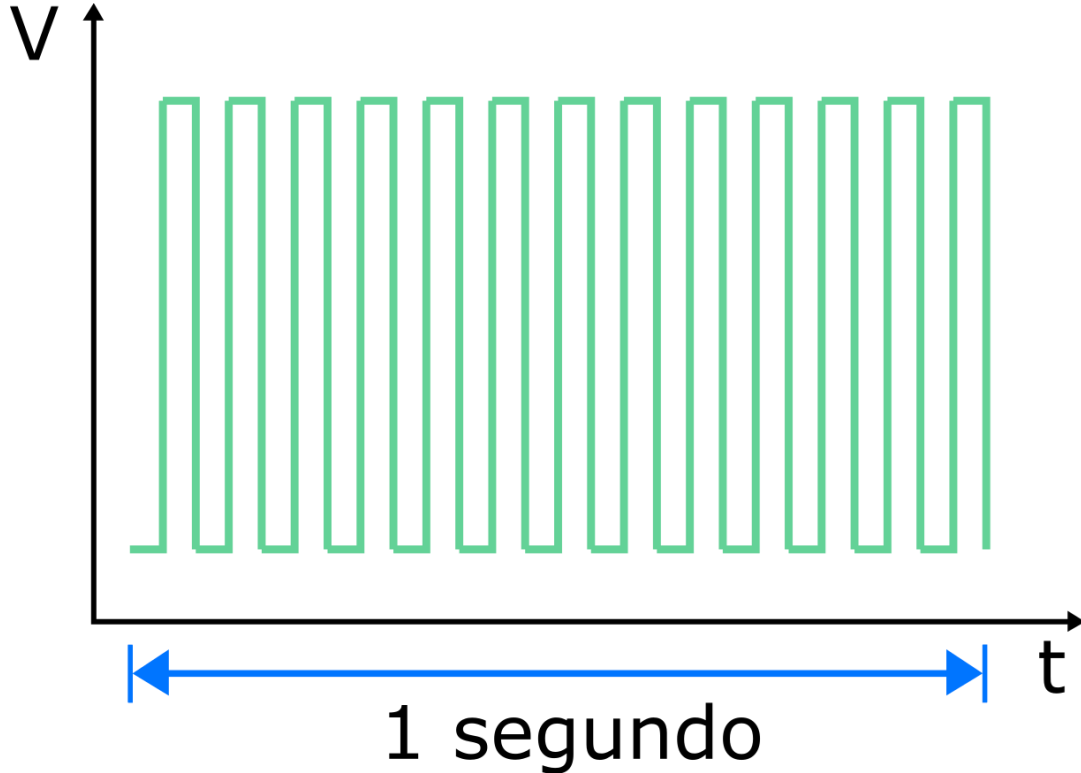
# Método de medición



Por Flancos: Conteo de ocurrencias en un tiempo determinado.



# Método de medición

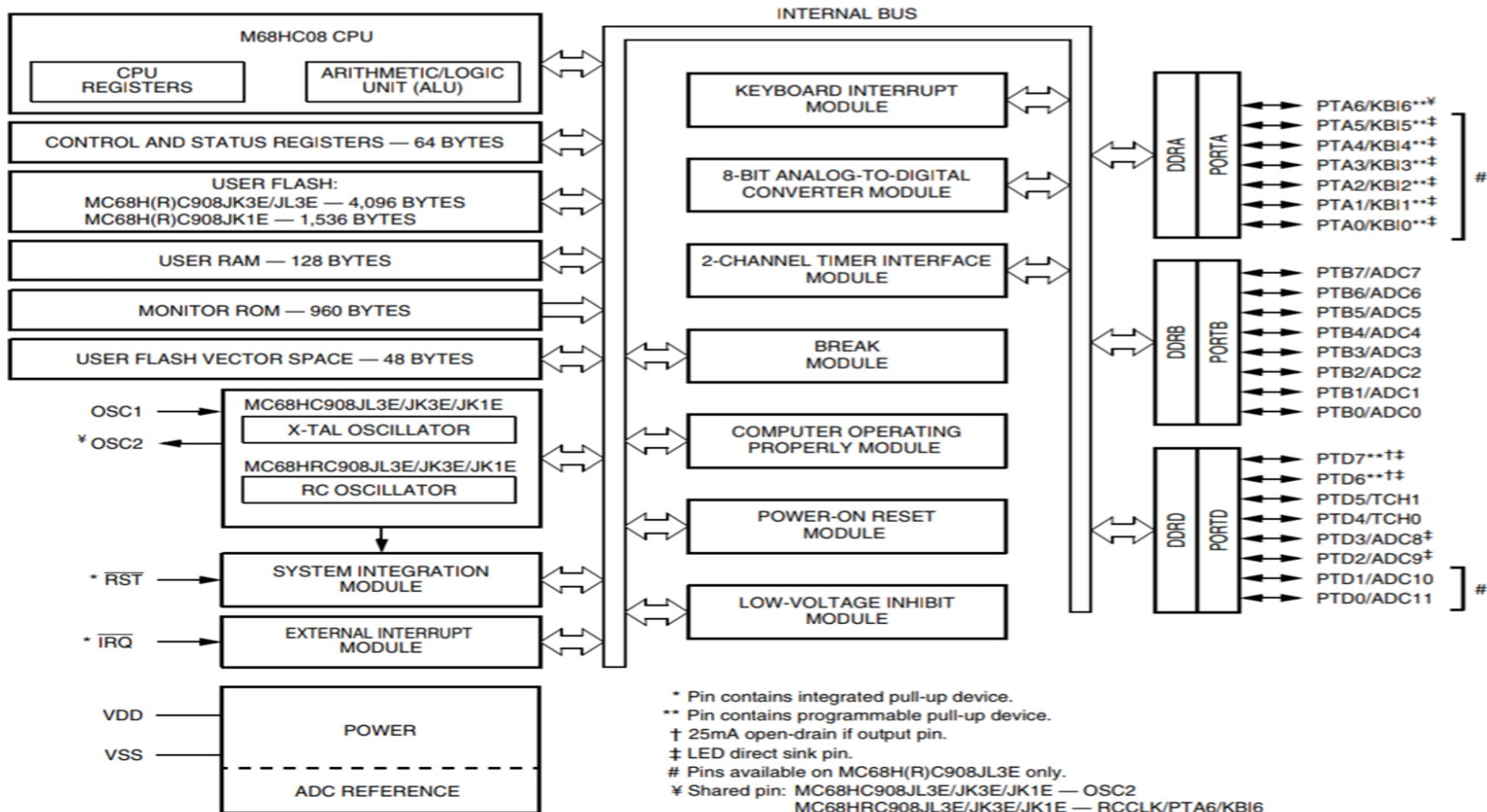


En una ventana de 1 segundo se contará la cantidad de flancos

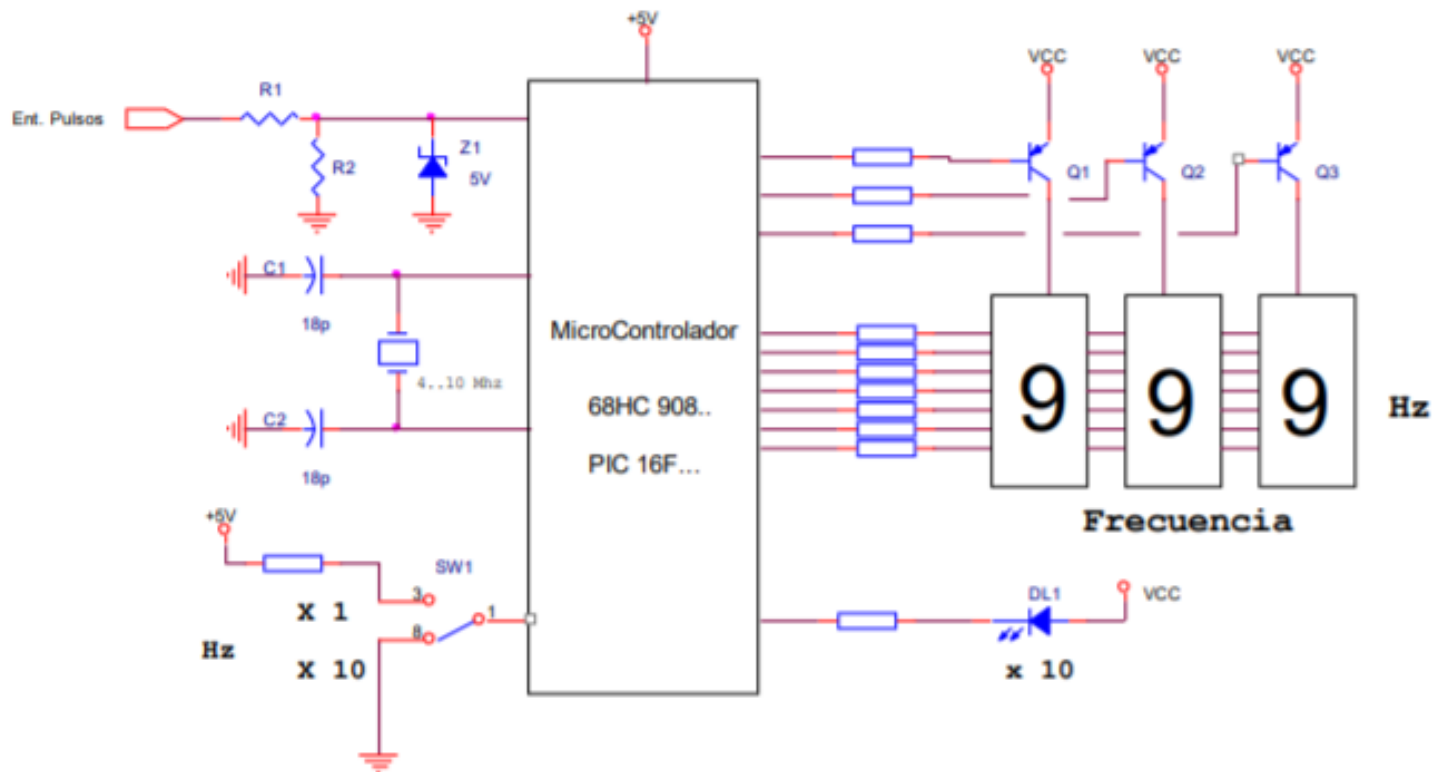
Ej.: 13 flancos contados en 1 segundo son 13 Hz

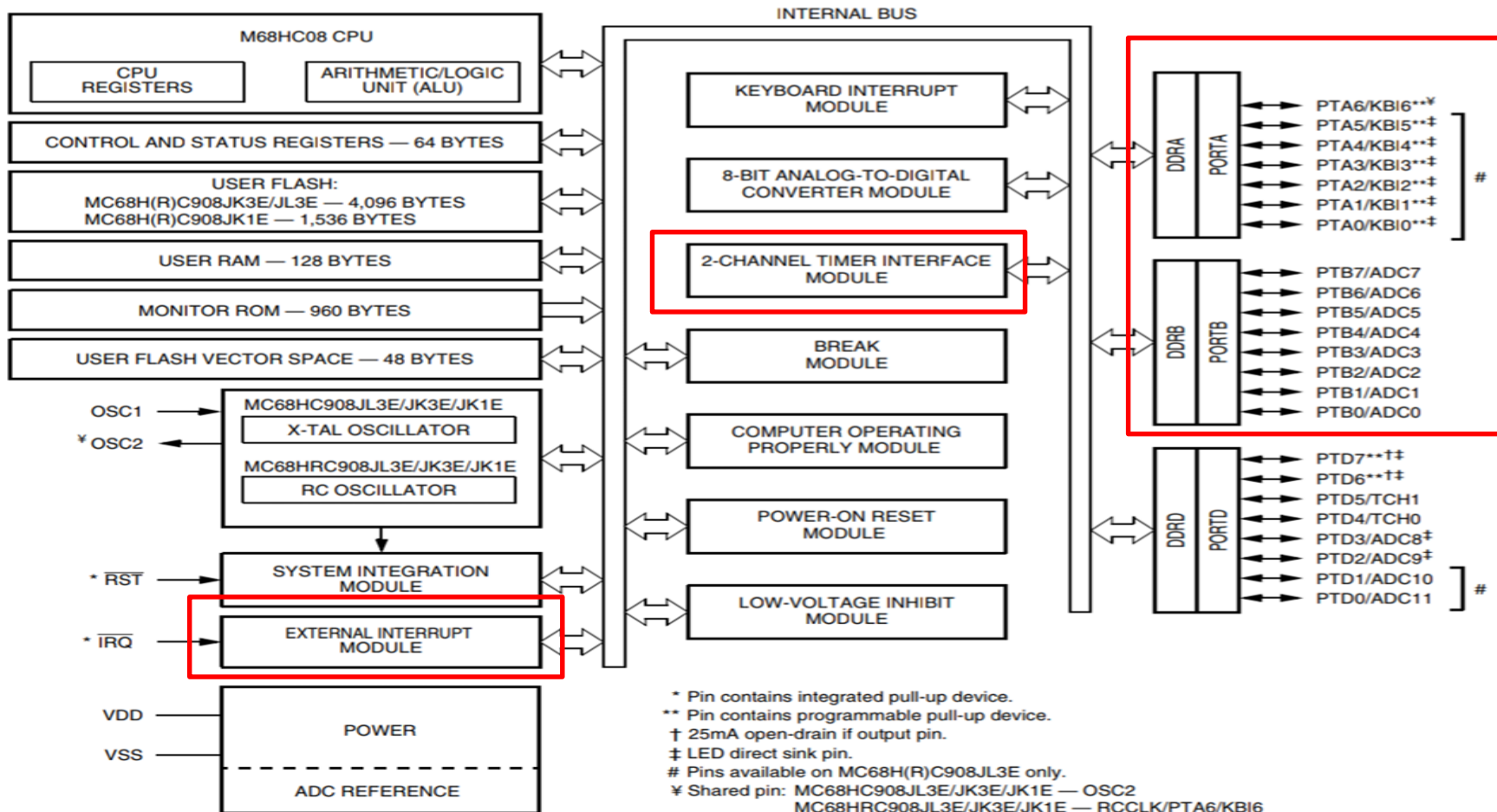
# Lectura de la hoja de datos

Para conocer el set de instrucciones y los dispositivos integrados dentro del microcontrolador, sus funciones y configuraciones.

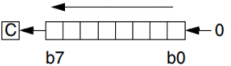


## Diagrama del hardware para esta aplicación.

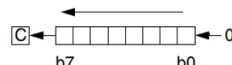




# Set de instrucciones

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z				
ADC #opr ADC opr ADC opr ADC opr,X ADC opr,X ADC ,X ADC opr,SP ADC opr,SP	Add with Carry	$A \leftarrow (A) + (M) + (C)$	†	†	-	†	†	IMM DIR EXT IX2 IX1 IX SP1 SP2	A9 B9 C9 D9 E9 F9 9EE9 9ED9	ii dd hh ll ee ff ff ff ff ee ff	2 3 4 4 3 2 4 5
ADD #opr ADD opr ADD opr ADD opr,X ADD opr,X ADD ,X ADD opr,SP ADD opr,SP	Add without Carry	$A \leftarrow (A) + (M)$	†	†	-	†	†	IMM DIR EXT IX2 IX1 IX SP1 SP2	AB BB CB DB EB FB 9EEB 9EDB	ii dd hh ll ee ff ff ff ff ee ff	2 3 4 4 3 2 4 5
AIS #opr	Add Immediate Value (Signed) to SP	$SP \leftarrow (SP) + (16 \ll M)$	-	-	-	-	-	IMM	A7	ii	2
AIX #opr	Add Immediate Value (Signed) to H:X	$H:X \leftarrow (H:X) + (16 \ll M)$	-	-	-	-	-	IMM	AF	ii	2
AND #opr AND opr AND opr AND opr,X AND opr,X AND ,X AND opr,SP AND opr,SP	Logical AND	$A \leftarrow (A) \& (M)$	0	-	-	†	†	IMM DIR EXT IX2 IX1 IX SP1 SP2	A4 B4 C4 D4 E4 F4 9EE4 9ED4	ii dd hh ll ee ff ff ff ff ee ff	2 3 4 4 3 2 4 5
ASL opr ASLA ASLX ASL opr,X ASL ,X ASL opr,SP	Arithmetic Shift Left (Same as LSL)		†	-	-	†	†	DIR INH INH IX1 IX SP1	38 48 58 68 78 9E68	dd ff ff ff ff ff	4 1 1 4 3 5

# Set de instrucciones

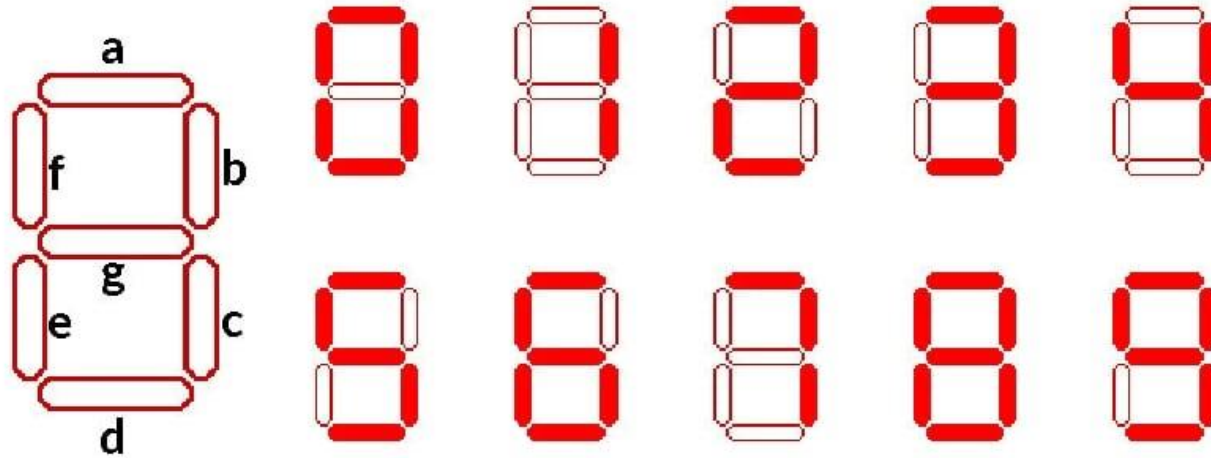
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z				
ADC #opr ADC opr ADC opr ADC opr,X ADC opr,X ADC ,X ADC opr,SP ADC opr,SP	Add with Carry	$A \leftarrow (A) + (M) + (C)$	†	†	-	†	†	IMM DIR EXT IX2 IX1 IX SP1 SP2	A9 B9 C9 D9 E9 F9 9EE9 9ED9	ii dd hh ll ee ff ff ff ff ee ff	2 3 4 4 3 2 4 5
ADD #opr ADD opr ADD opr ADD opr,X ADD opr,X ADD ,X ADD opr,SP ADD opr,SP	Add without Carry	$A \leftarrow (A) + (M)$	†	†	-	†	†	IMM DIR EXT IX2 IX1 IX SP1 SP2	AB BB CB DB EB FB 9EEB 9EDB	ii dd hh ll ee ff ff ff ff ee ff	2 3 4 4 3 2 4 5
AIS #opr	Add Immediate Value (Signed) to SP	$SP \leftarrow (SP) + (16 \ll M)$	-	-	-	-	-	IMM	A7	ii	2
AIX #opr	Add Immediate Value (Signed) to H:X	$H:X \leftarrow (H:X) + (16 \ll M)$	-	-	-	-	-	IMM	AF	ii	2
AND #opr AND opr AND opr AND opr,X AND opr,X AND ,X AND opr,SP AND opr,SP	Logical AND	$A \leftarrow (A) \& (M)$	0	-	-	†	†	IMM DIR EXT IX2 IX1 IX SP1 SP2	A4 B4 C4 D4 E4 F4 9EE4 9ED4	ii dd hh ll ee ff ff ff ff ee ff	2 3 4 4 3 2 4 5
ASL opr ASLA ASLX ASL opr,X ASL ,X ASL opr,SP	Arithmetic Shift Left (Same as LSL)		†	-	-	†	†	DIR INH INH IX1 IX SP1	38 48 58 68 78 9E68	dd ff ff ff ff ff	4 1 1 4 3 5

# **Display 7 segmentos**

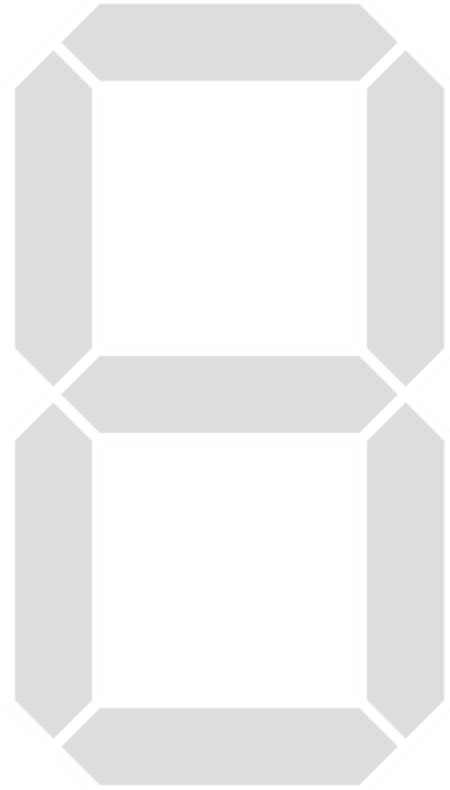
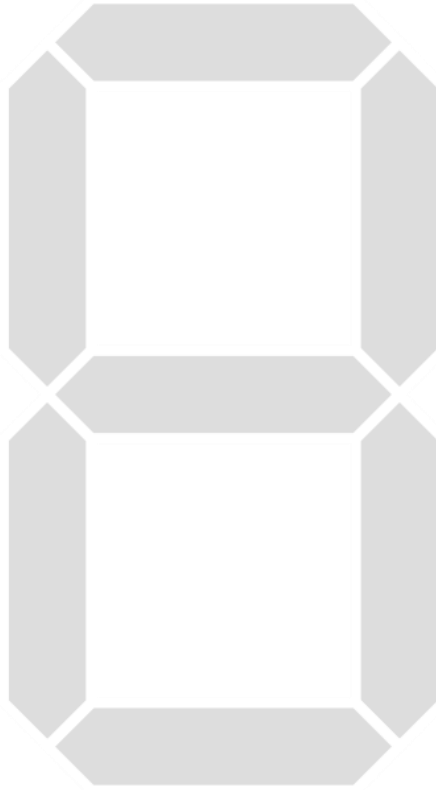
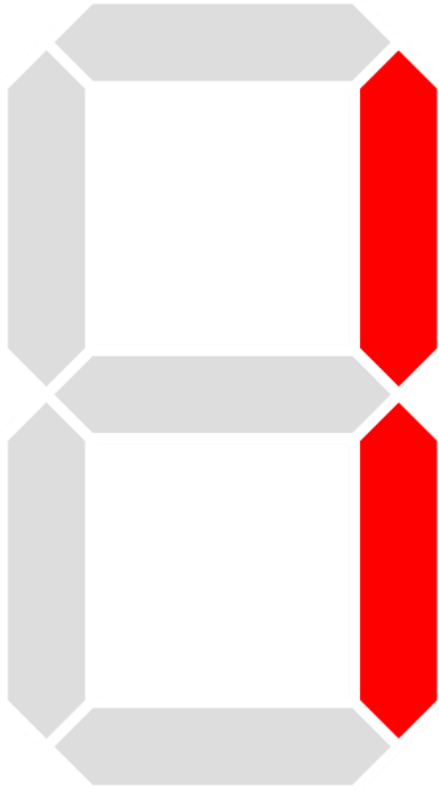
Módulo con 7 leds que  
permiten formar  
caracteres  
alfanuméricos



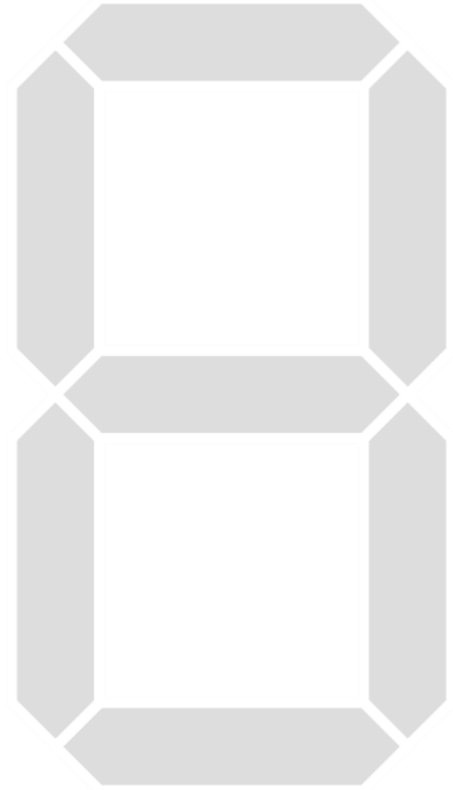
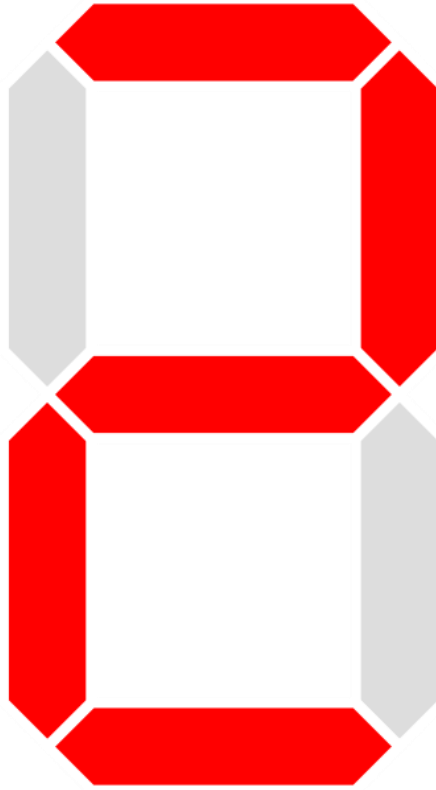
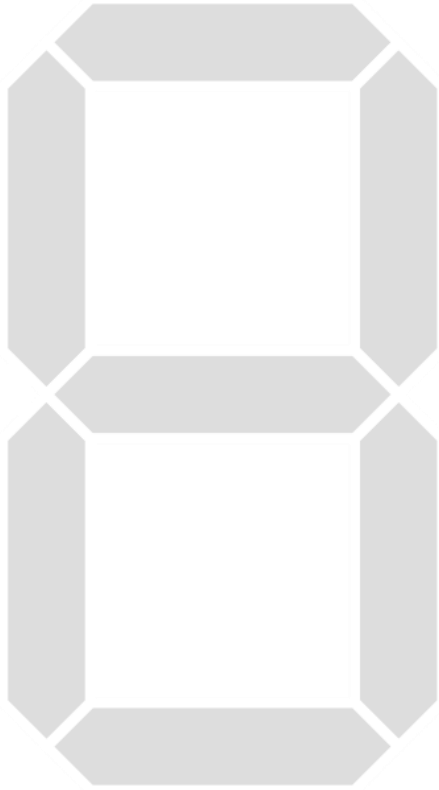
# Display de 7 segmentos



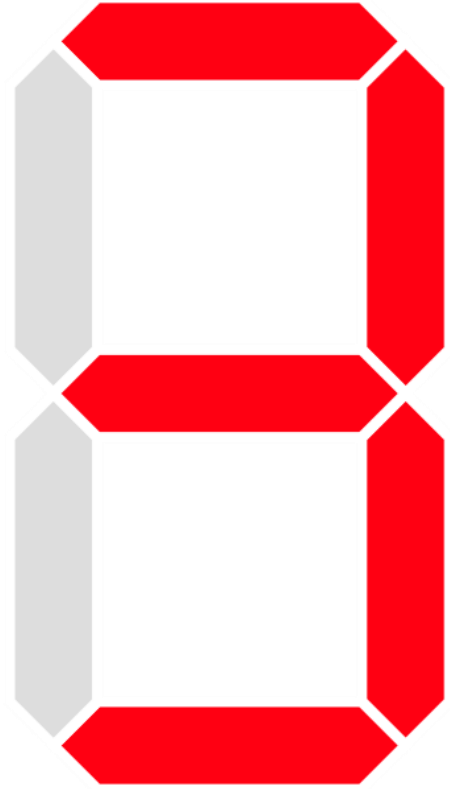
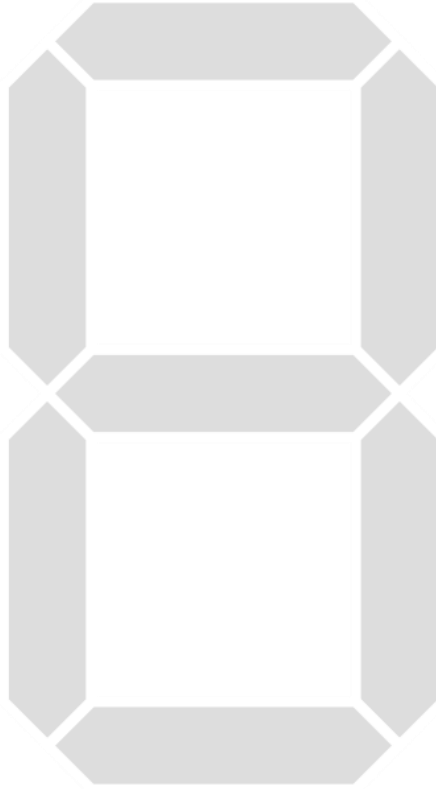
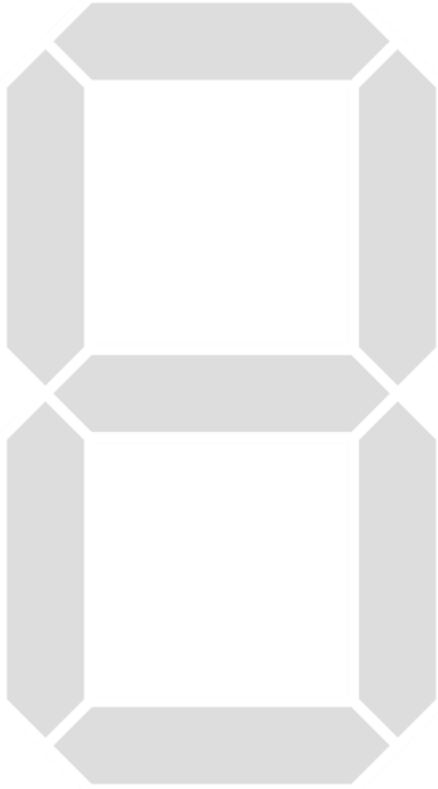
# Multiplexación



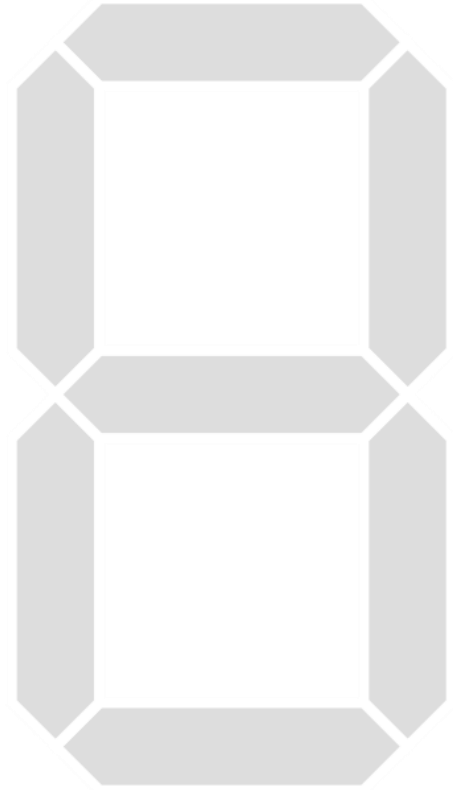
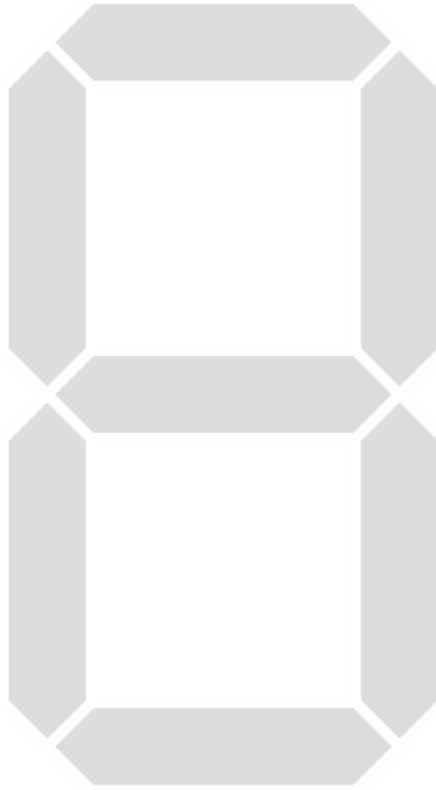
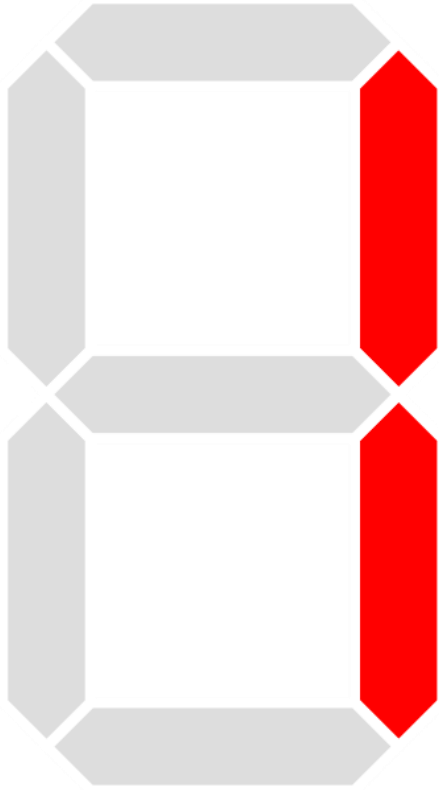
# Multiplexación



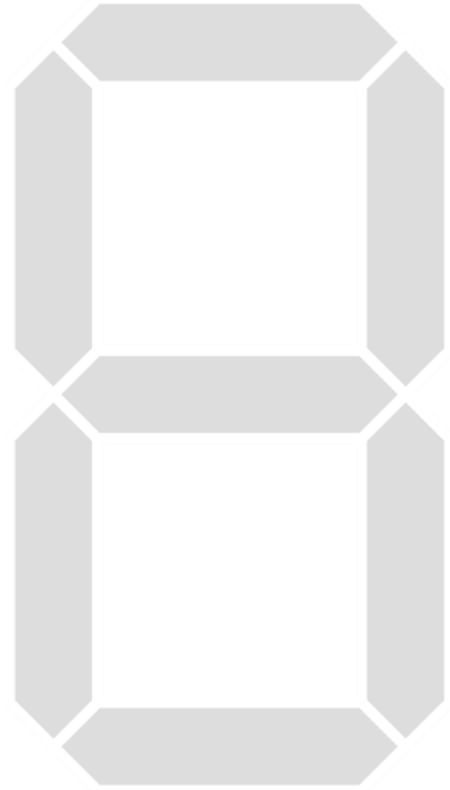
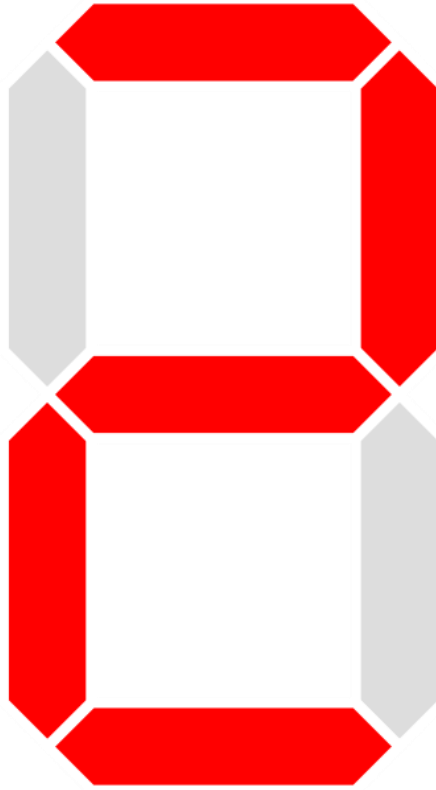
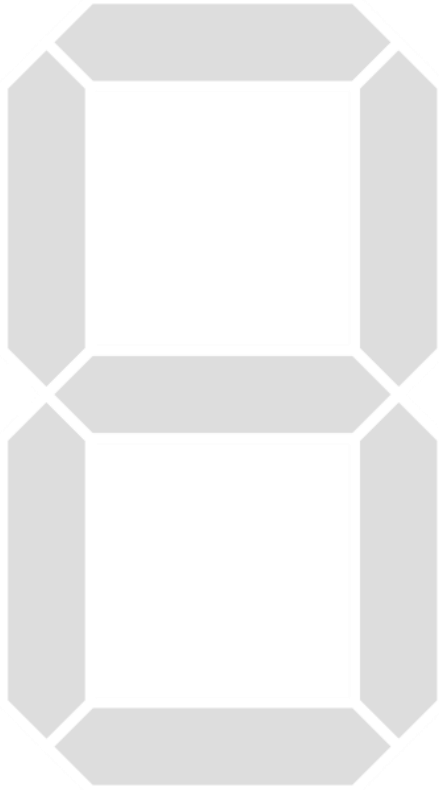
# Multiplexación



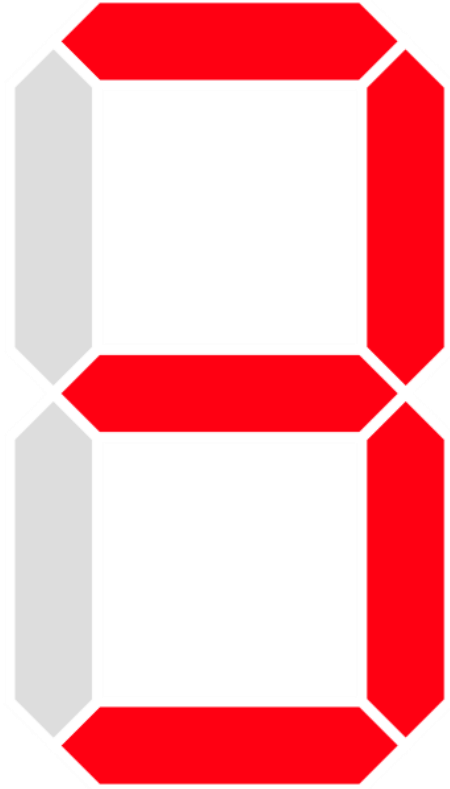
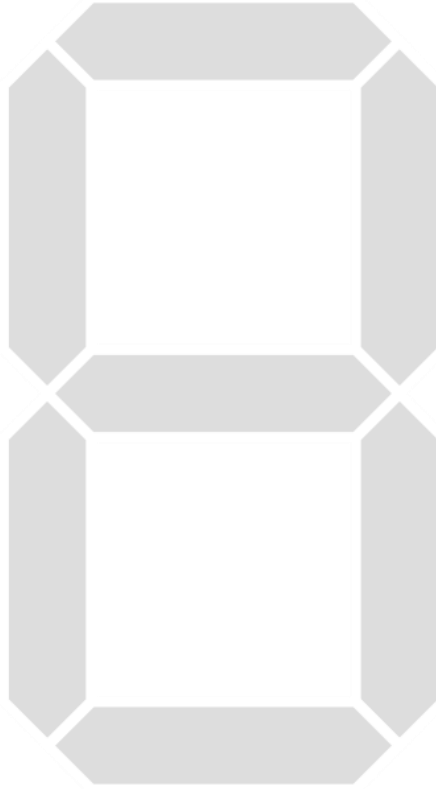
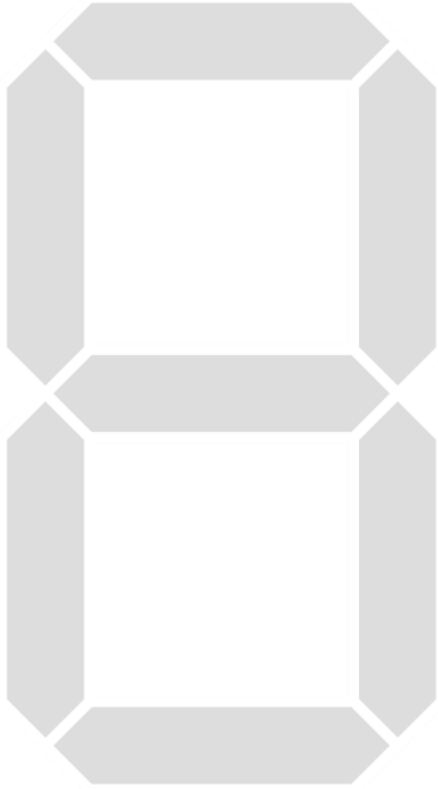
# Multiplexación



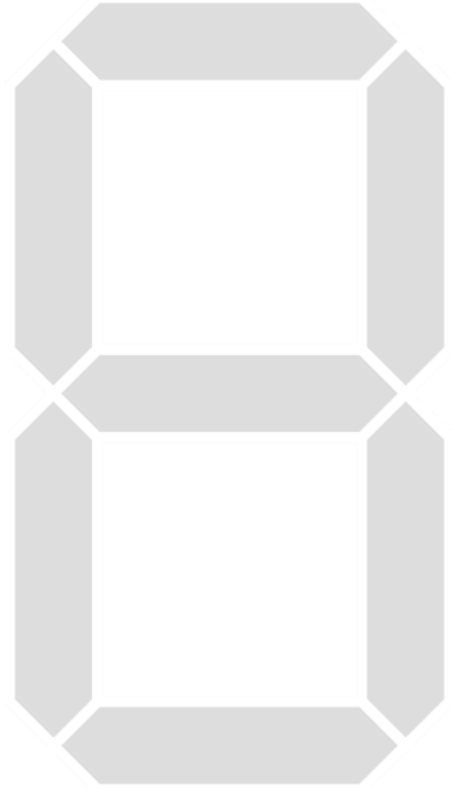
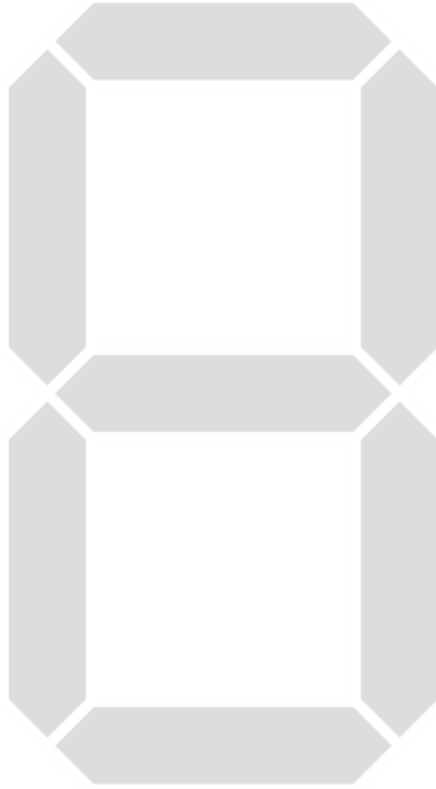
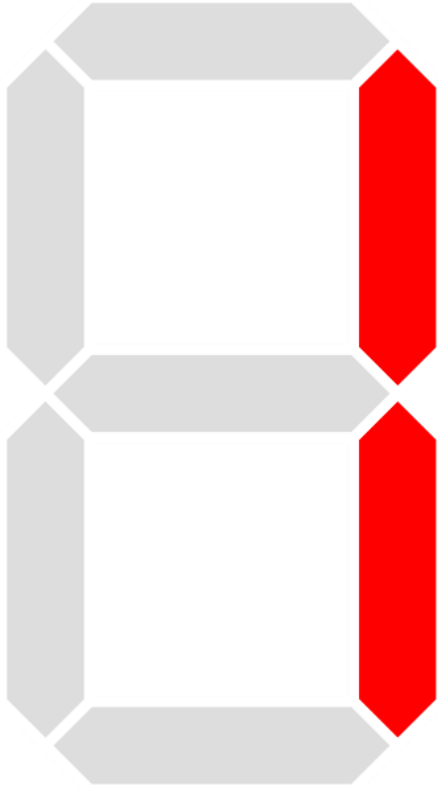
# Multiplexación



# Multiplexación

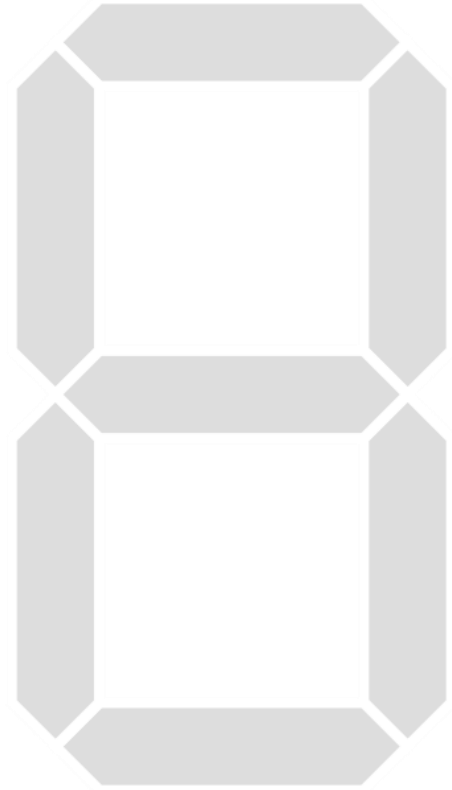
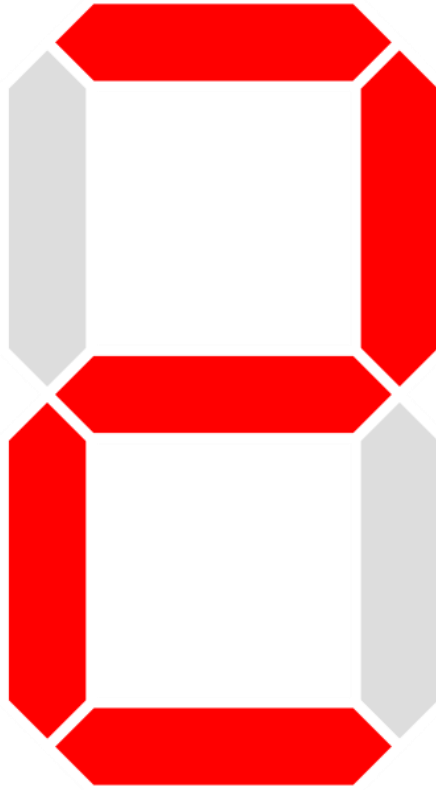
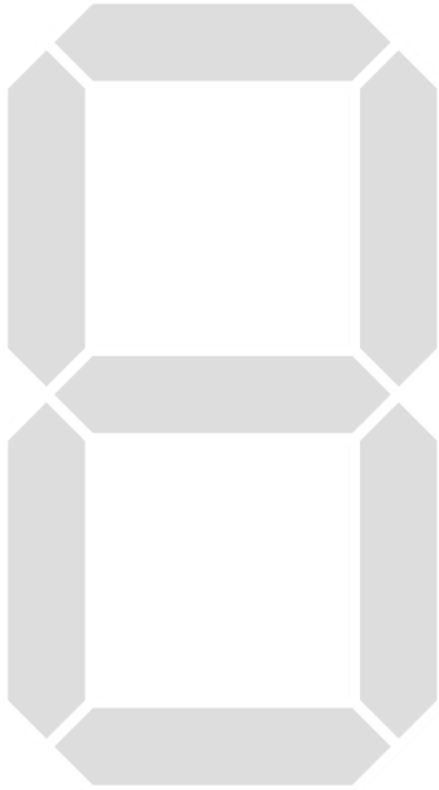


# Multiplexación

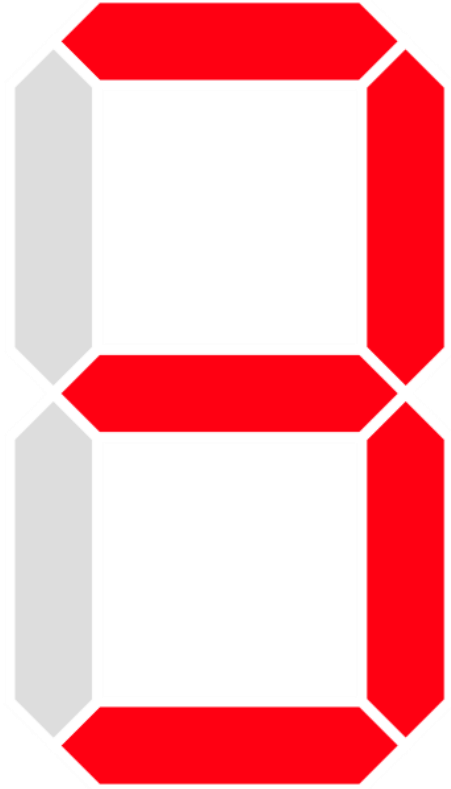
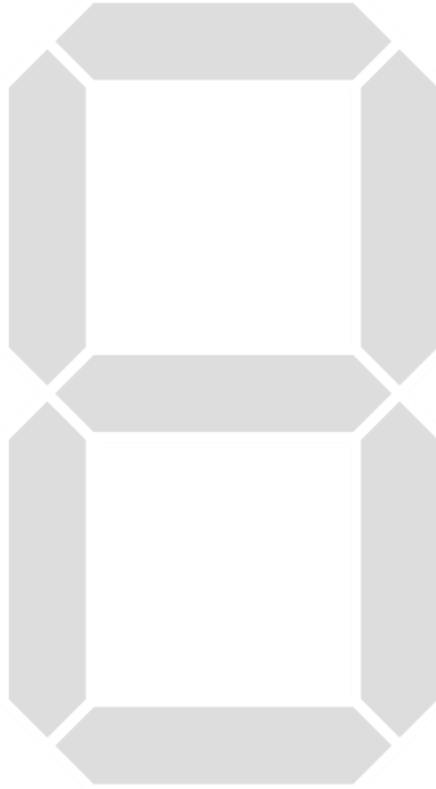
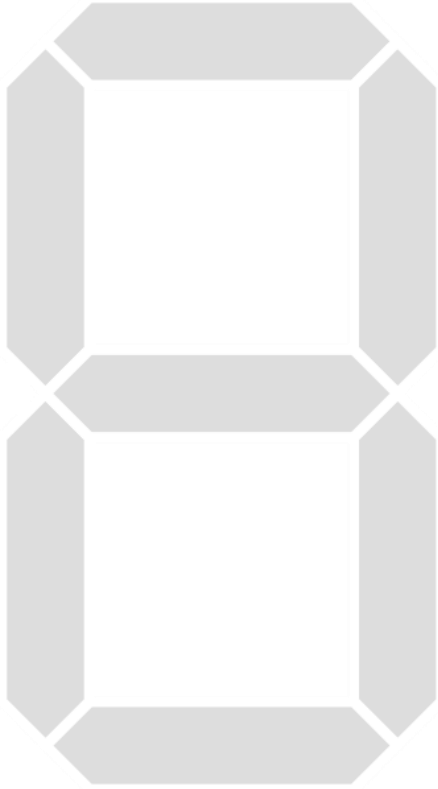




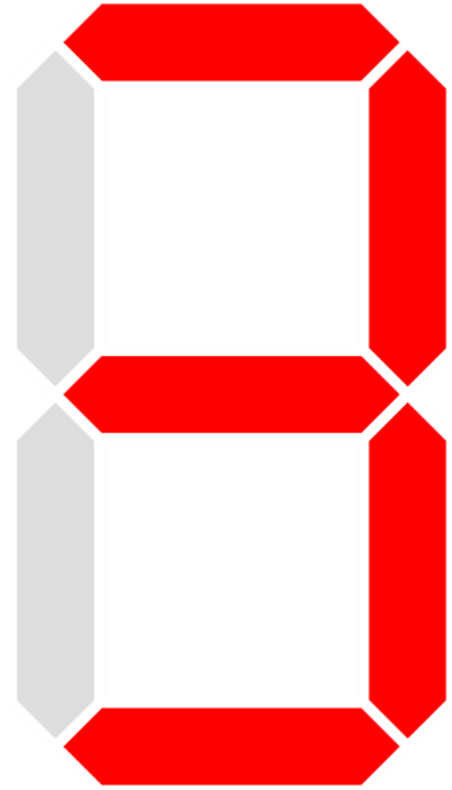
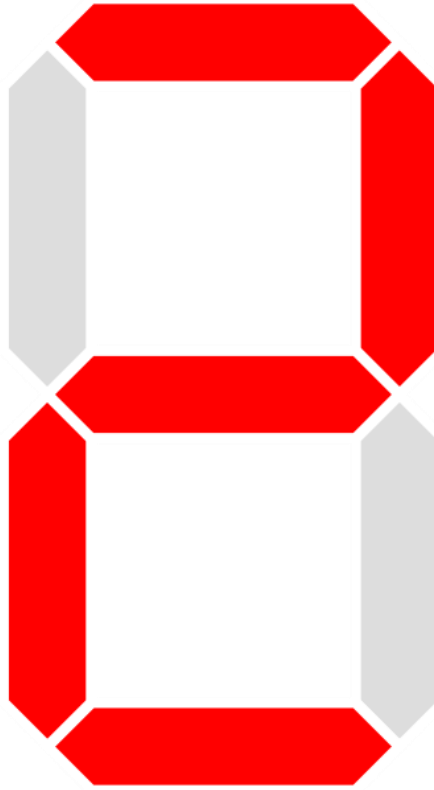
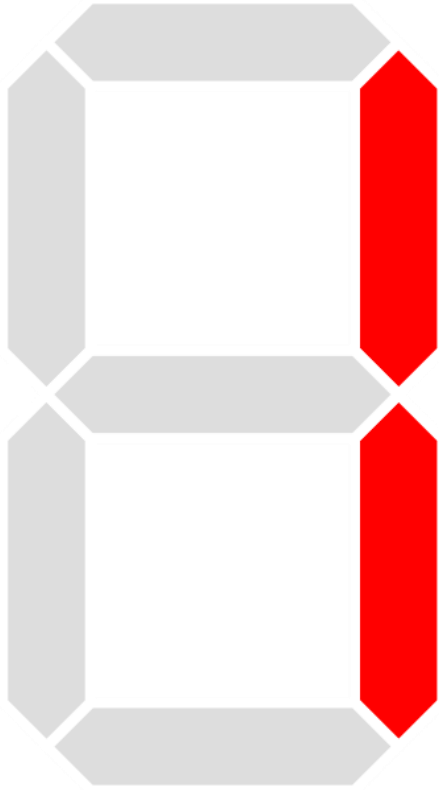
# Multiplexación



# Multiplexación



# Multiplexación



# **Interrupciones**

Son las que permiten realizar tareas para eventos específicos.

# Función de las Interrupciones

Cuando una interrupción ocurre, el procesador deja de realizar su tarea principal y atiende a la rutina especificada por el evento de interrupción ocurrido.

# Fuentes de interrupción

- El microcontrolador a utilizar permite generar interrupciones periódicas mediante un contador interno, el cual se puede configurar. **(Timer)**
- También se puede generar una interrupción externa mediante una conexión física. **(Flancos de la señal)**

# Adquisición de los datos

- Conteo de flancos en sistema BCD de 4 dígitos (millar, centena, decena y unidad)
-

# Sistema BCD

- Codificación de dígitos decimales en binario
- Cuatro dígitos binarios representarán uno decimal.

$$0000 = 0$$

$$0001 = 1$$

$$0010 = 2$$

....

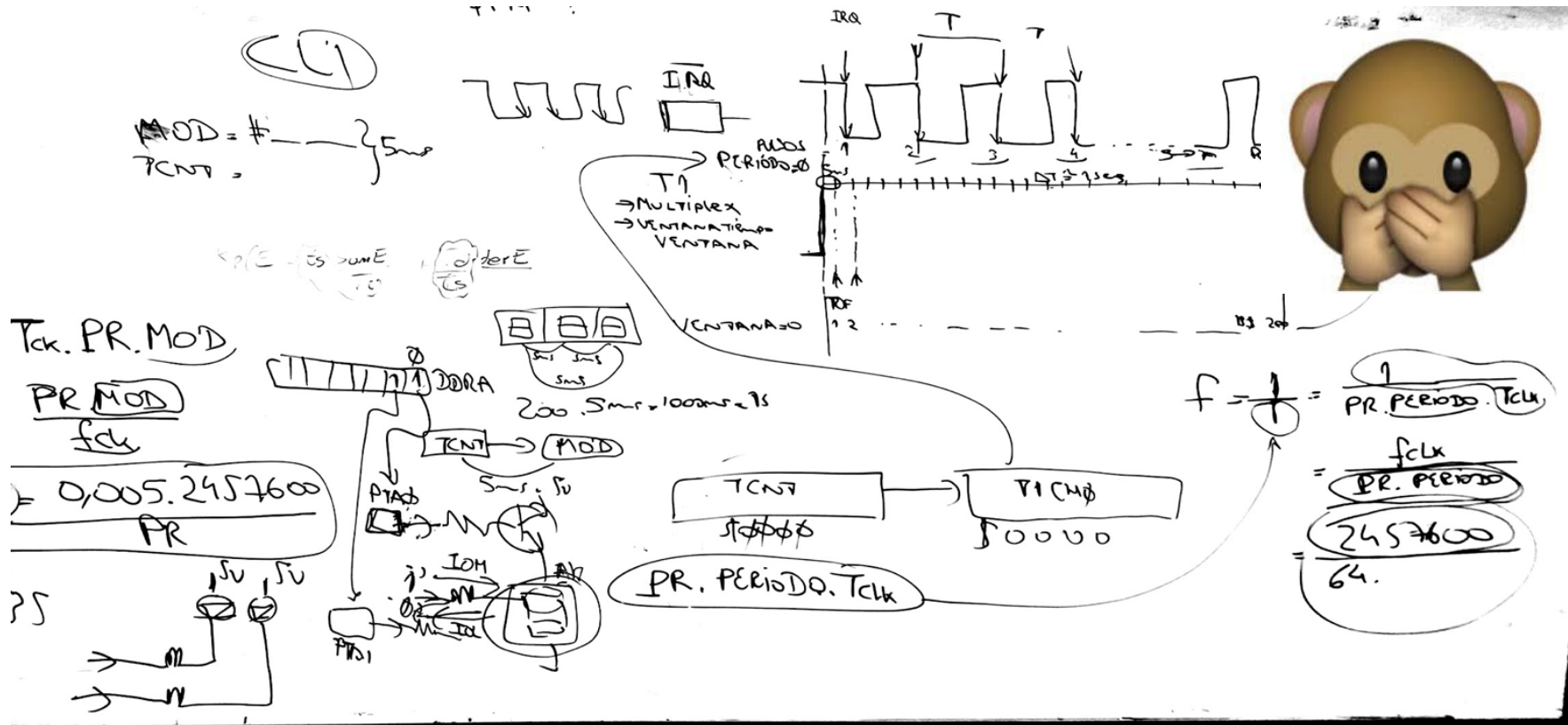
$$1000 = 8$$

$$1001 = 9$$



Al contar en BCD no  
es necesaria una  
conversión

# Primeras ideas

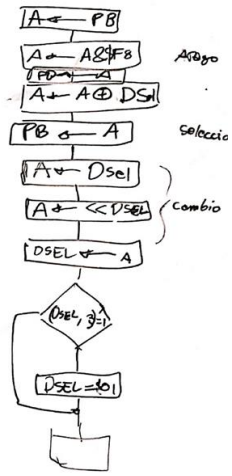
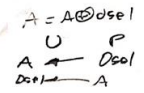
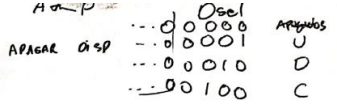
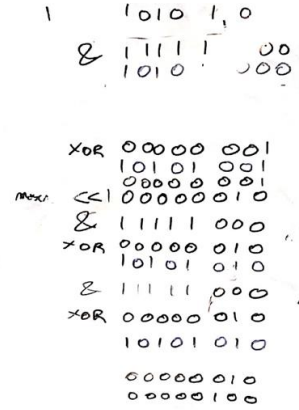
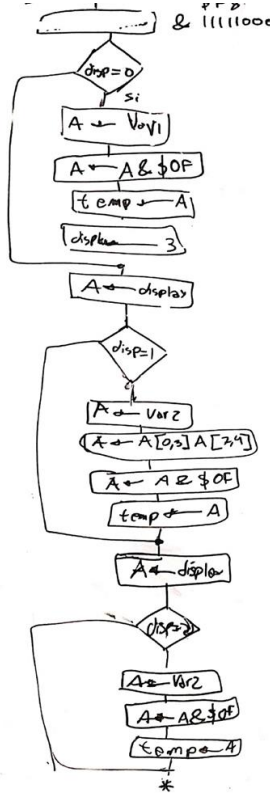
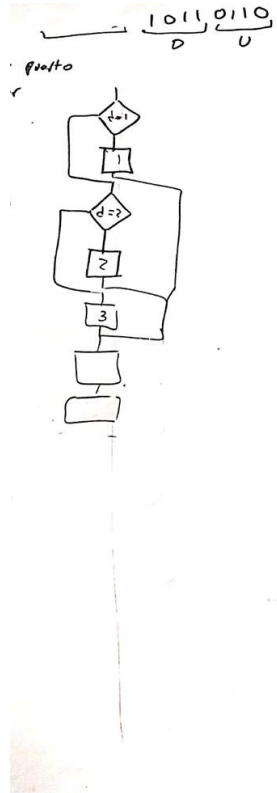


# Diagramas de flujo

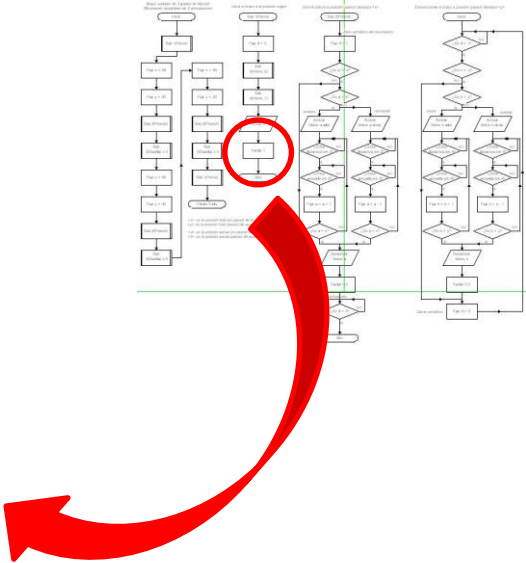
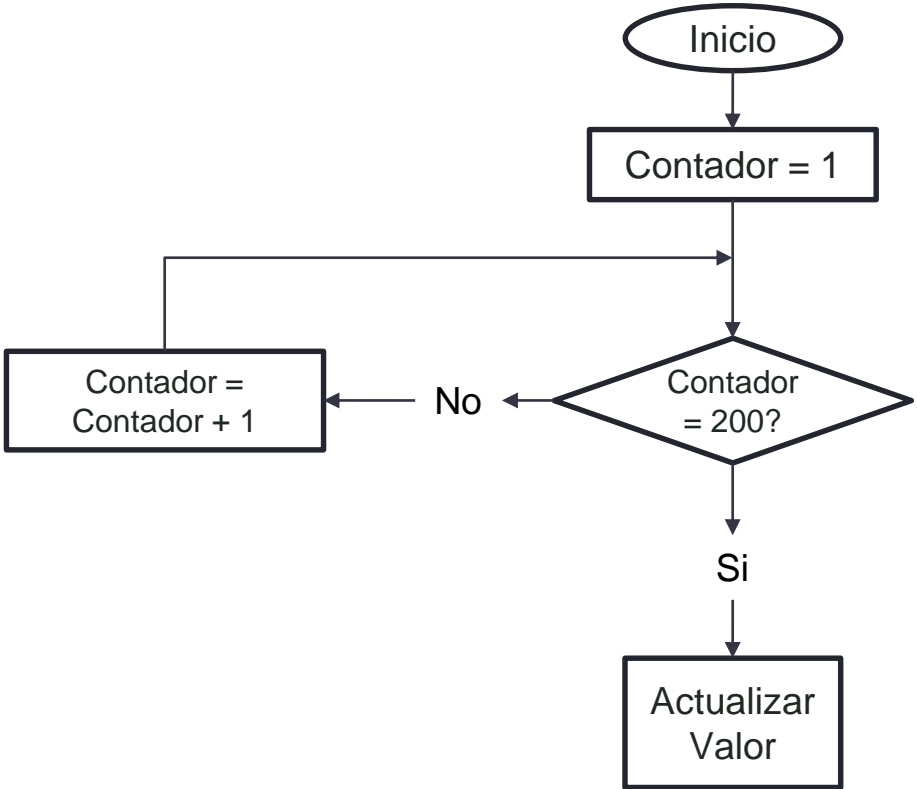
¿Para qué utilizamos un diagrama de flujo?

- Es una guía de las acciones y decisiones que el programa debe realizar
  - Se traducen al lenguaje a utilizar y las instrucciones del procesador utilizado
-

# Primeros Diagramas de Flujo



# Diagrama de flujo del contador



```

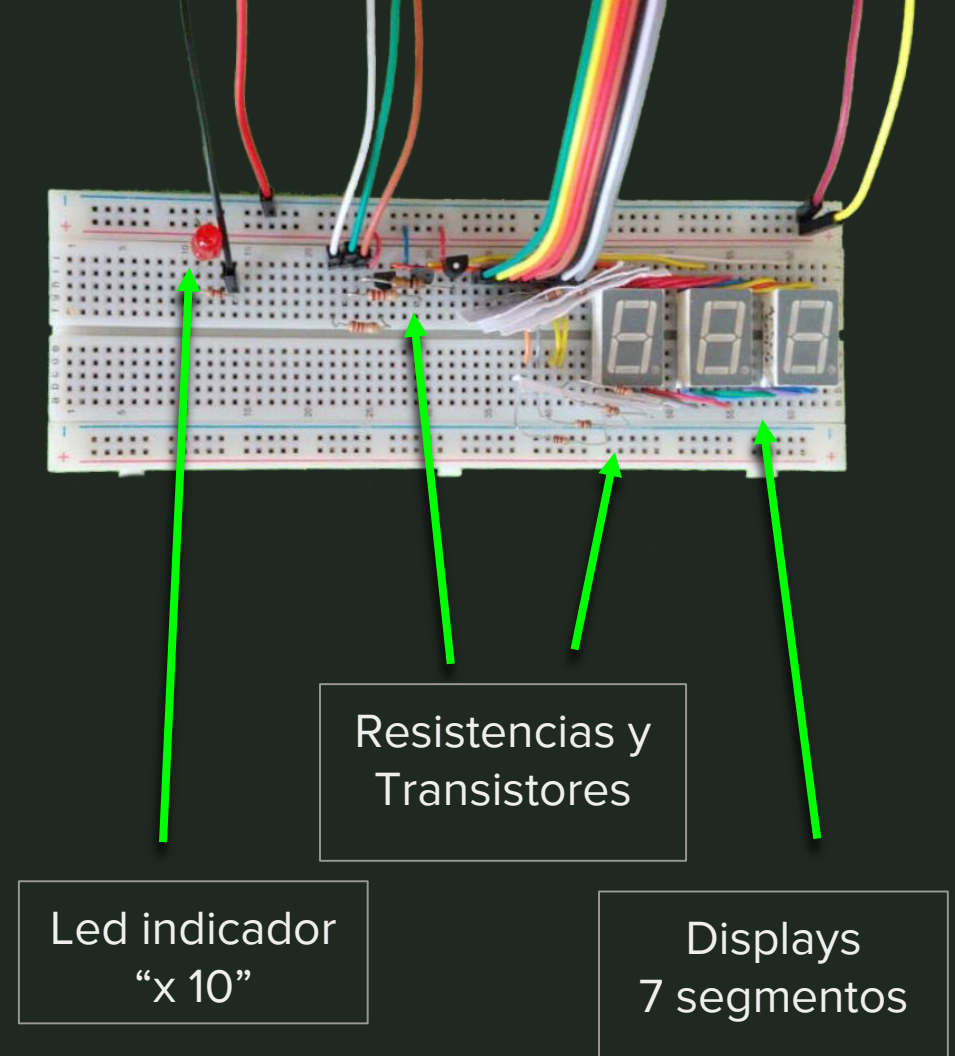
128 ;-----
129 ;/TOF/TOIE/TSTOP/TRST/----/PS2/PS1/PS0/
130 ;-----
131 ;TOF =1 FLAG DE INTERRUPCIÓN DEL TIMER.
132 ;TOIE =1 HABILITA INTERUPCIONES POR OVERFLOW DEL TIMER
133 ;TSTOP=0 HABILITA EL CONTADOR FREE RUNNING
134 ;TRST=1 RESET DEL MODULO
135 ;PS[2:0] CONFIGURA EL PRESCALER
136 ;-----
137     STA     TSC
138     RTS
139
140 KBCD27 LDA     RESULT+2
141         AND     #$0F
142         TAX
143         CLRH
144         |      LDA     TABLA,X
145         STA     DECENA
146         LDA     RESULT+2
147         NSA
148         AND     #$0F
149         TAX
150         LDA     TABLA,X
151         STA     CENTENA
152         |      INC CENTENA
153         LDA     RESULT+3
154         NSA
155         AND     #$0F
156         TAX
157         LDA     TABLA,X
158         STA     UNIDAD
159         RTS
160

```



# Prueba en Protoboard

Se realiza una primera implementación







# Muchas gracias!!

¿Preguntas?

