

## COMUNICACIÓN PARALELA

Puertos A, B, E y K.

Características comunes:

- Bidireccionales
- Programables pin a pin como entrada o salida.
- Tienen un registro para habilitar o deshabilitar “pull-resistor”<sup>1</sup>
- Registro para habilitar o deshabilitar salida en baja potencia.
- Registro para programar modos de operación.

A continuación se presenta un diagrama a bloques de lo que es un pin de un puerto en el HC12:

---

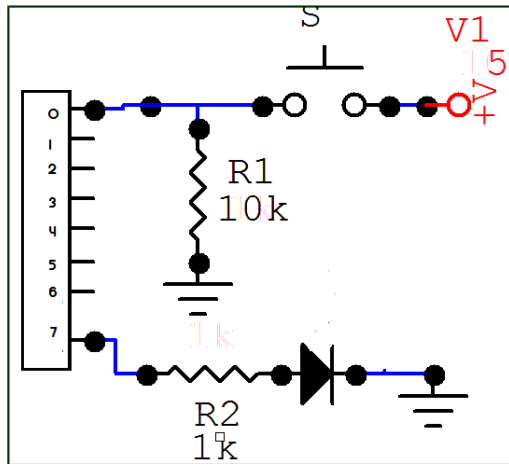
<sup>1</sup> Pull-up resistor: resistencia conectada a Vcc  
pull-down resistor: resistencia conectada a tierra.

Las direcciones de algunos puertos se presentan a continuación:

PORTA	\$0000
PORTB	\$0001
DDRA	\$0002
DDRB	\$0003

**EJEMPLO:** encendido de un led vía interruptor de contacto momentáneo.

Función and.

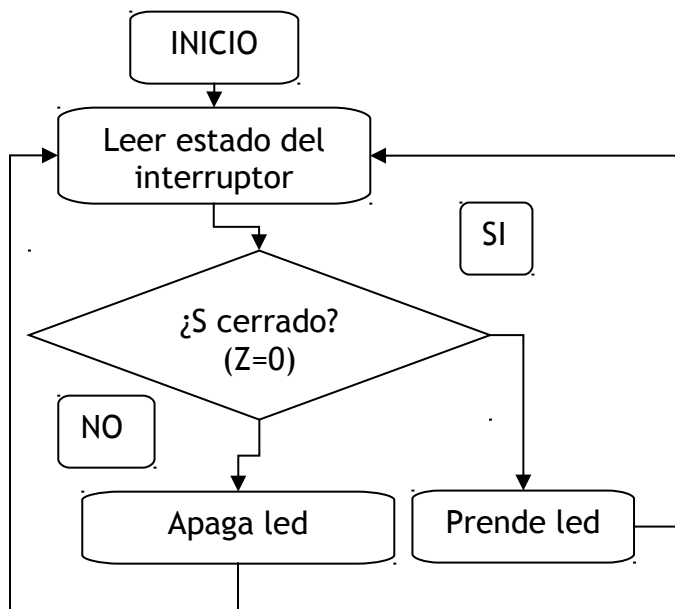


*	*	*	*	*	*	*	0
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0

Resultado = 0 => interruptor abierto (Z=1)

*	*	*	*	*	*	*	1
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1

Resultado = 1 => interruptor cerrado (Z=0)



PORTA	EQU	\$00	;define pseudo operadores
DDRA	EQU	\$02	; define pseudo operador
	LDX	#\$0000	; carga el índice X de manera inmediata con 0000
	LDAA	#\$80	; carga el acumulador A con 1000 0000h, bit7 a '1'
	STAA	DDRA, X	; almacénalo en el registro DDRA
REV	LDAA	PORTA, X	; carga el acumulador A con lo que lee en el puerto A
	ANDA	#\$01	; operación and, A con 01
	BNE	PRENDE	; salta si Z=0 (a la etiqueta PRENDE
	CLRA		
	STAA	PORTA, X	
	BRA	REV	
PRENDE	LDAA	#\$80	
	STAA	PORTA, X	
	BRA	REV	

Otra forma seria programándolo con la instrucción BSET

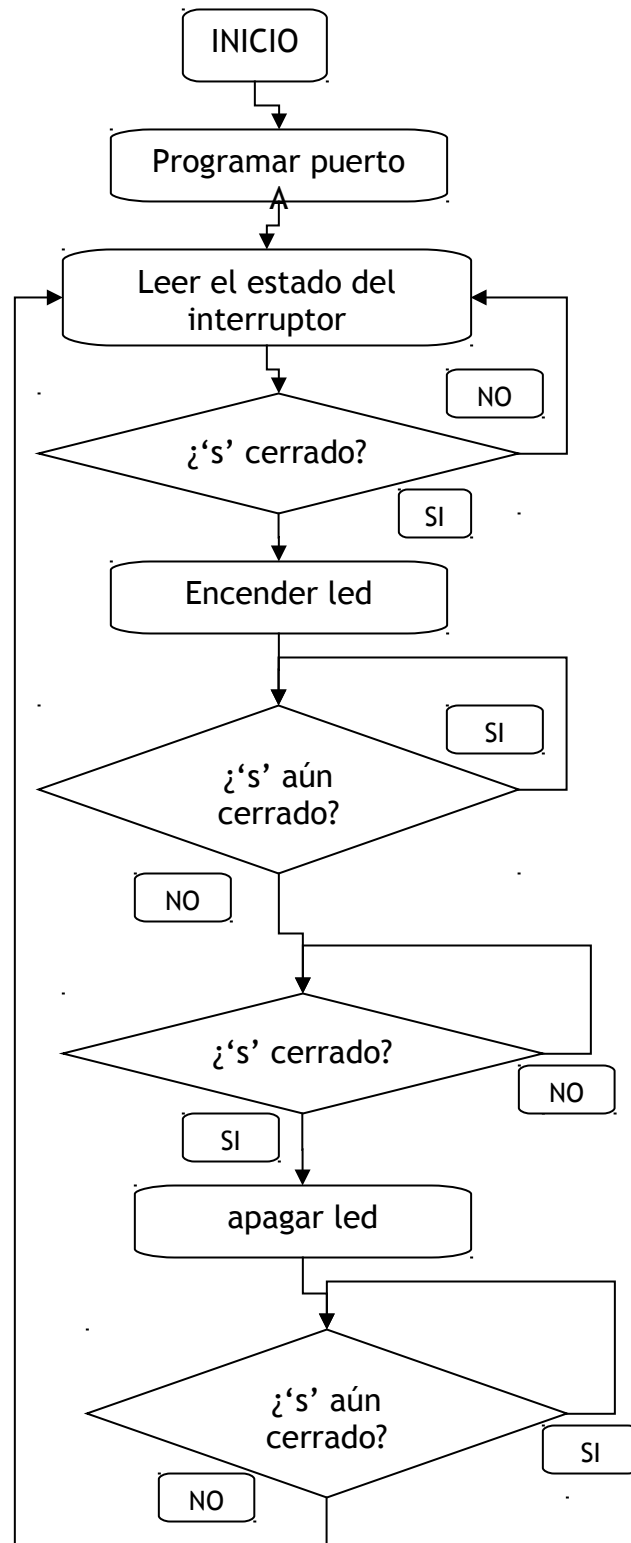
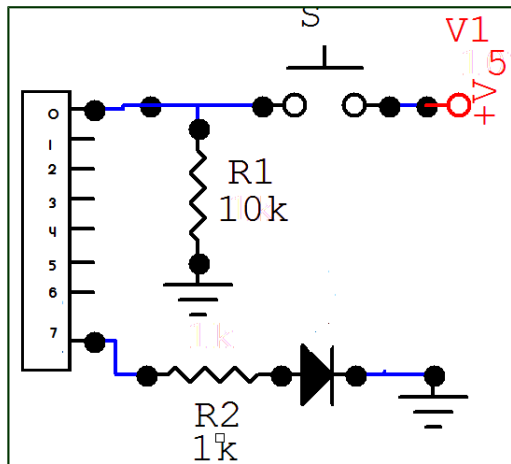
PORTA	EQU	\$00	;define pseudo operadores
DDRA	EQU	\$02	; define pseudo operador
	LDX	#\$0000	; carga el índice X de manera inmediata con 0000
	BSET	DDRA,X,\$80	; bit set pone un '1' en el bit7 del registro DDRA de manera indexada. <b>OJO, ya que esta instrucción solo pone un '1' en el bit que le indiquemos, los demás bits quedan sin cambios</b>
REV	LDAA	PORTA, X	; carga el acumulador A con lo que lee en el puerto A
	ANDA	#\$01	; operación and, A con 01
	BNE	PRENDE	; salta si Z=0 a la etiqueta PRENDE
	CLRA		
	STAA	PORTA, X	
	BRA	REV	
PRENDE	LDAA	#\$80	
	STAA	PORTA, X	
	BRA	REV	

Otra manera seria como sigue, las instrucciones se reducen:

PORTA	EQU	\$00	;define pseudo operadores
DDRA	EQU	\$02	; define pseudo operador
	LDX	#\$0000	; carga el índice X de manera inmediata con 0000
	BSET	DDRA,X,\$80	; bit set pone un '1' en el bit7 del registro DDRA de manera indexada. <b>OJO, ya que esta instrucción solo pone un '1' en el bit que le indiquemos, los demás bits quedan sin cambios</b>
	BRSET	PORTA,X,\$01, <b>PRENDE</b>	; Salta a la etiqueta PRENDE si hay un '1' en el bit0 del puerto A, de manera indexada.
	BCLR	PORTA,X,\$80	; pone un '0' en el bit7 del puerto A de manera indexada.
	BRA	REV	; salta a la etiqueta REV
<b>PRENDE</b>	BSET	PORTA,X,\$80	; pone un '1' en el bit7 del puerto A de manera indexada.
	BRA	REV	

## PROGRAMA (TOGGLE)

Encender el led y apagar el led cada vez que se presione una vez el interruptor. El puerto A es el que se muestra:



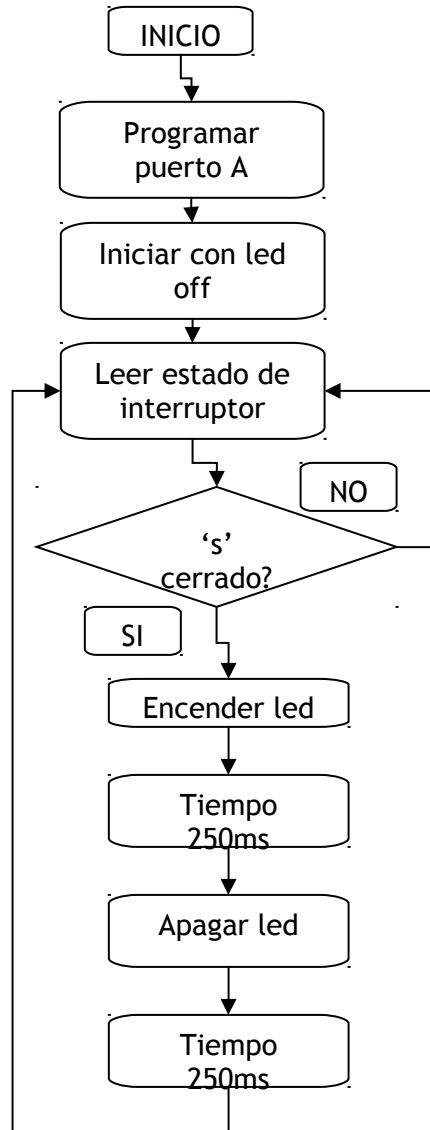
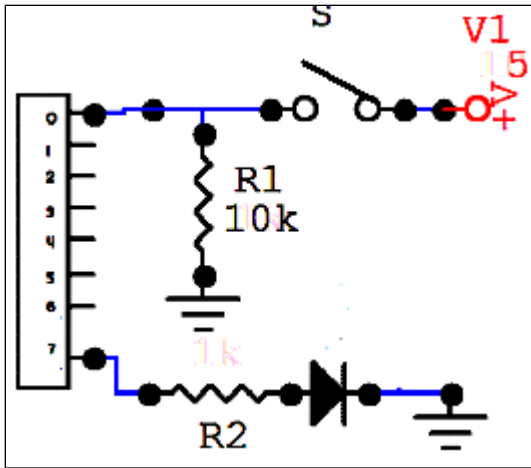
## CODIGO<sup>2</sup>

PORTA	EQU	\$00	;define pseudo operadores
DDRA	EQU	\$02	; define pseudo operador
	LDX	#\$0000	; carga el índice X de manera inmediata con 0000
	BSET	DDRA,X,\$80	; bit set pone un '1' (entrada) en el bit7 del registro DDRA de manera indexada. <b>OJO, ya que esta instrucción solo pone un '1' en el bit que le indiquemos, los demás bits quedan sin cambios</b>
	BCLR	DDRA,X,\$01	; bit clear pone un '0' (salida) en el bit0 del registro DDRA de manera indexada.
LEER	BRCLR	PORTA,X,\$01,LEER	; Salta si hay un '0' a la etiqueta LEER en el bit0 del puerto A, de manera indexada.
	LDAA	#\$80	
	STAA	PORTA,X	
AUN	BRSET	PORTA,X,\$01,AUN	; Salta si hay un '1' a la etiqueta AUN en el bit0 del puerto A, de manera indexada.
CERRADO	BRCLR	PORTA,X,\$01,CERRADO	; Salta si hay un '0' a la etiqueta CERRADO en el bit0 del puerto A, de manera indexada.
	LDAA	#\$00	
	STAA	PORTA,X	
AUN2	BRSET	PORTA,X,\$01, AUN2	; Salta si hay un '0' a la etiqueta AUN2 en el bit0 del puerto A, de manera indexada.
	BRA	LEER	; salta a la etiqueta leer.

<sup>2</sup> Cada ciclo (~) es típicamente de 125[ns] para un bus de 8[MHz] (oscilador de 16[MHz])

## PROBLEMA

Desarrollar un programa que haga parpadear un led cuando el interruptor 'S' de contacto sostenido se cierre.



PORTA EQU \$00  
DDRA EQU \$02  
LDX #\$0000  
LDAA #\$80  
STAA DDRA,X

;define pseudo operadores  
;define pseudo operador

INICIA BCLR PORTA,X,\$80  
LEER BRCLR PORTA,X\$01,INICIA  
BSET PORTA,X,\$80

JSR RETARDO (4)  
BCLR PORTA,X,\$80  
JSR RETARDO  
BRA LEER

Para programar el retardo observe:

FFFFh=65535d

Por 5 ciclos veces que se realiza el lazo, se Realizan un LDY, un DECA y BNE, el cual suman 6 ciclos de instrucción.

RETARDO	LDAA	#\$Δ	(1)
LAZO	LDY	#\$FFFF	(2)
SUBLAZO	NOP		(1)
	DEY		(1)
	BNE	SUBLAZO	(3)
	DECA		(1)
	BNE	LAZO	(3)
	RTS		(5)

$4_{JSR} + 1_{LDAA} + [6 + 5(65535)]Δ + 5_{RTS}$   
ESTO DEBEMOS IGUALARLO A:

$250\ 000\ [Ms] = \{4_{JSR} + 1_{LDAA} + [6 + 5(65535)]Δ + 5_{RTS}\}0.125[Ms]$

$Δ = \{(250000/0.125) - 4 - 1 - 5\} / (6 + 5(65535))$

Δ=6.10347 en decimal

POR LO TANTO:

**Δ=6h**

Si deseamos 50[ms]



$$\Delta = \{(50000/0.125) - 4 - 1 - 5\} / (6 + 5(65535))$$
$$\Delta = 1.220$$

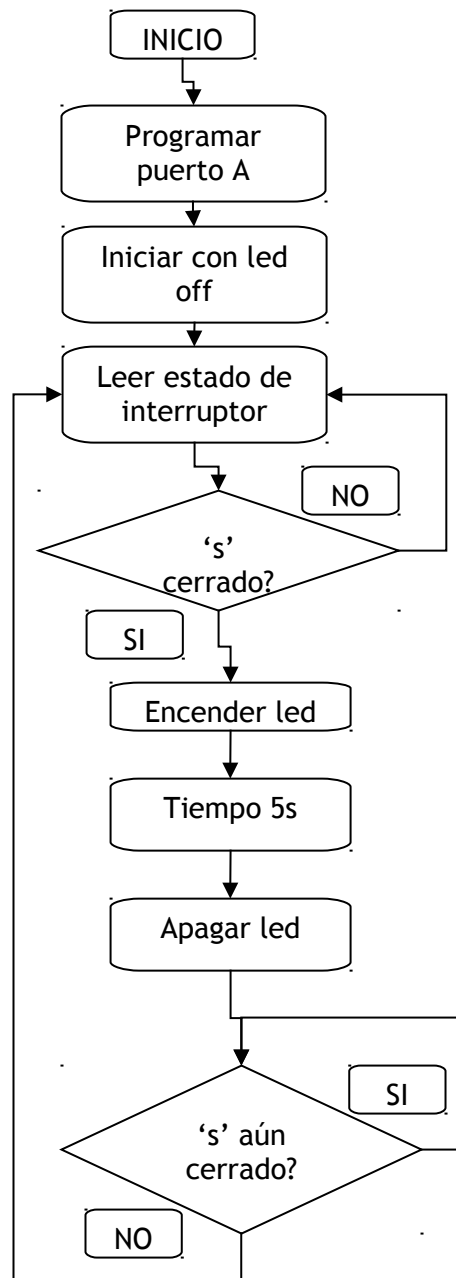
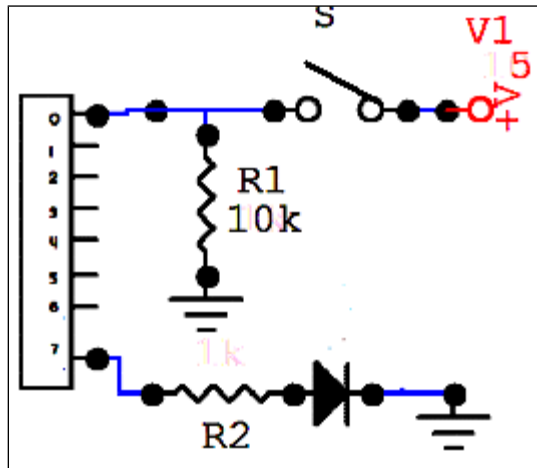
$$\Delta = 1H$$

¿Qué pasa si deseamos tener un retardo de 50 [ms]?, hay diferentes formas de hacerlo, como nos damos cuenta es la quinta parte del tiempo de retardo del ejemplo anterior, por lo que el cálculo quedaría de la siguiente manera:

$$[Ms] = \{4_{JSR} + 1_{LDAA} + [6 + (65535)]\Delta + 5_{RTS}\}0.125[Ms]$$

## PROBLEMA

Desarrollar un programa que encienda el led durante 5 segundos cada vez que se cierra el interruptor 's' de contacto sostenido.



PORTA

EQU

\$00

;define pseudo operadores

DDRA EQU \$02  
LDX #\$0000  
LDAA #\$80  
STAA DDRA,X

; define pseudo operador

LEER BCLR PORTA,X,\$80  
BRCLR PORTA,X,\$01,LEER  
BSET PORTA,X,\$80

JSR RETARDO5S (4)  
BCLR PORTA,X,\$80  
AUN BRSET PORTA,X,\$01,AUN  
BRA LEER

Para programar el retardo observe:

FFFFh=65535d

Por 5 ciclos veces que se realiza el lazo, se Realizan un LDY, un DECA y BNE, el cual suman 6 ciclos de instrucción.

RETARDO 5S	LDAA	#\$Δ	(1)
LAZO	LDY	#\$FFFF	(2)
SUBLAZO	NOP		(1)
	DEY		(1)
	BNE	SUBLAZO	(3)
	DECA		(1)
	BNE	LAZO	(3)
	RTS		(5)

$4_{JSR} + 1_{LDAA} + [6 + 5(65535)]Δ + 5_{RTS}$   
ESTO DEBEMOS IGUALARLO A:

$5\ 000\ 000\ [Ms] = \{4_{JSR} + 1_{LDAA} + [6 + 5(65535)]Δ + 5_{RTS}\}0.125[Ms]$

$Δ = \{(5000000/0.125) - 4 - 1 - 5\} / (6 + 5(65535))$

Δ=122.07 en decimal

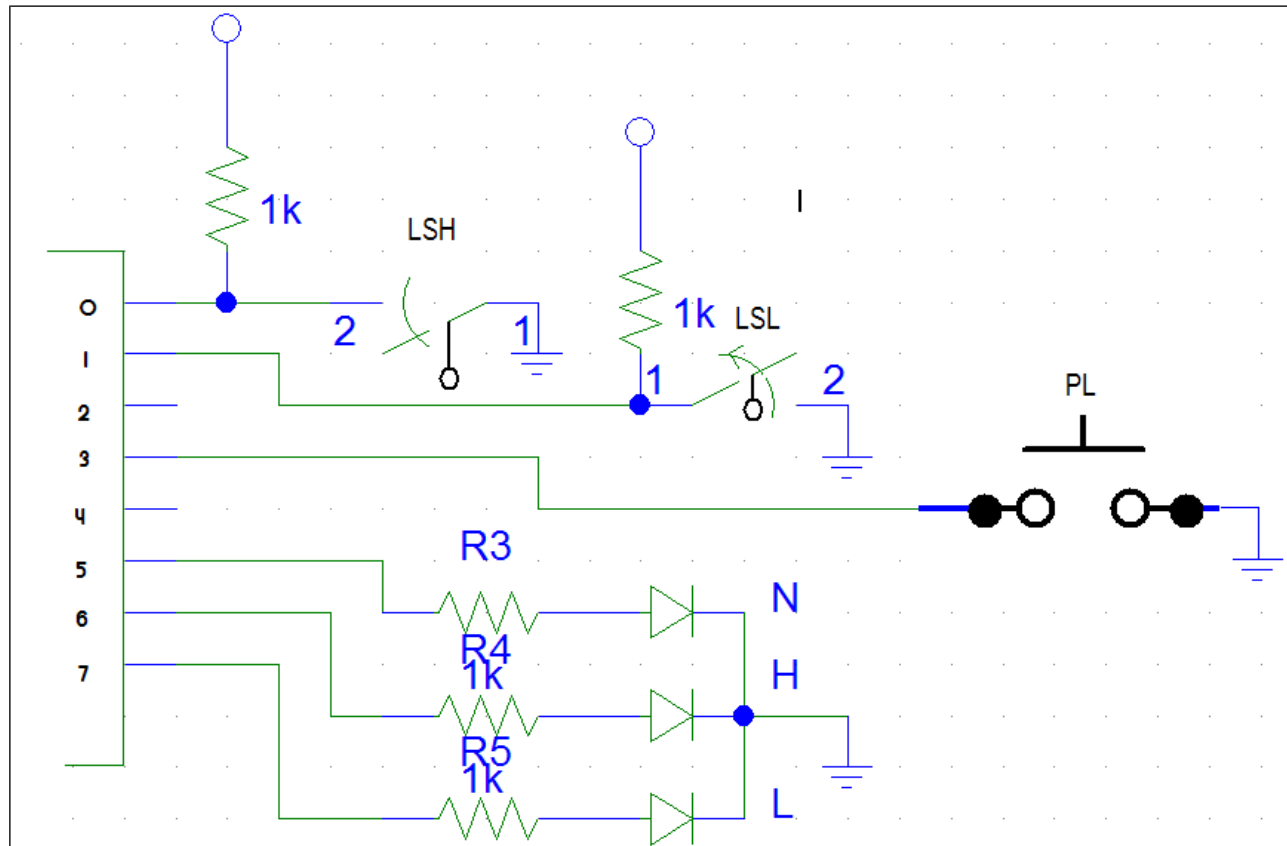
POR LO TANTO:

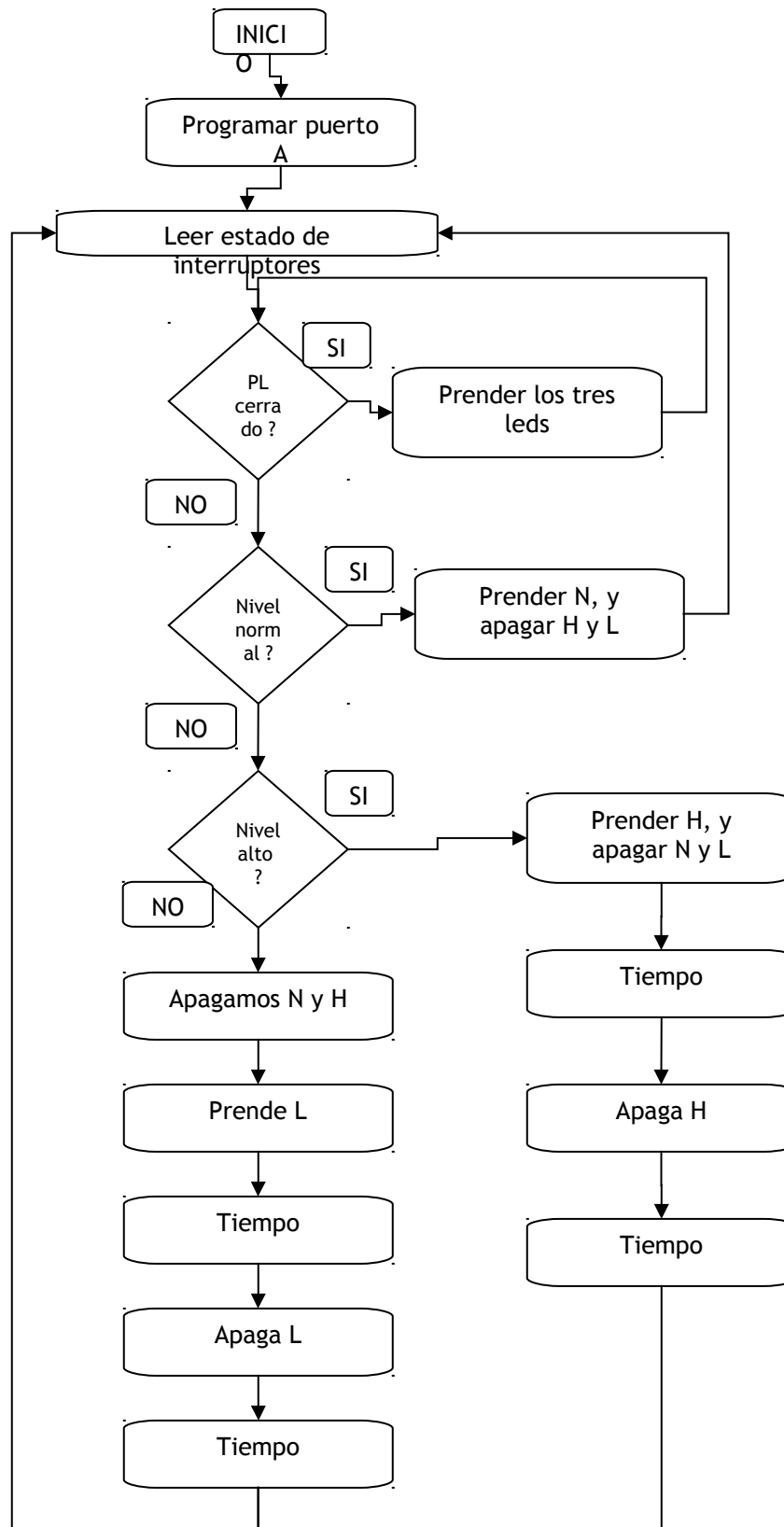
**Δ=7Ah**

En caso de que no nos alcanzara el conteo del acumulador A, utilizaríamos alternativamente el acumulador B, pero esto a su vez, implicaría ocupar todos los registros disponibles para programar un solo retardo, estos ejercicios se realizan para practicar con las instrucciones del hc12, más sin embargo, el microcontrolador cuenta con contadores específicos, que se verán más adelante.

## PROBLEMA

Realizar un programa que genere señales de alarma independientes por nivel alto o nivel bajo en el tanque, mediante el parpadeo de los leds correspondientes, si el nivel es normal, el led N debe prender, en condiciones de alarma se debe apagar, (prueba de leds), se cierra en cualquier tiempo y condición, los tres leds deben prender y permanecer así, hasta que el interruptor PL se abra para regresar al estado anterior.





PORTA	EQU	\$00	;define pseudo operadores
DDRA	EQU	\$02	; define pseudo operador
	LDX	#\$0000	
	LDAA	#\$E0	;programa puerto A
	STAA	DDRA,X	;programa puerto A
LEER	LDAA	PORTA,X	;carga el acum, con el contenido de A
REG	BRSET	PORTA,X,\$08,PRUEBA	;
	ANDA	#\$03	; AND con 03h y modifica A
	LDAA	PORTA,X	; volvemos a cargar A, debido a que ANDA lo modifica
	CMPA	#\$03	;
	BEQ	NORMAL	; salta a NORMAL, si (A)-03=0; Z=1
	BRCLR	PORTA,X,\$01,ALTO	
	BCLR	PORTA,X,\$60	; apaga N y H
	BSET	PORTA,X,\$80	
	JSR	RETRASO	
	BCLR	PORTA,X,\$80	
	JSR	RETRASO	; (4)
	BRA	LEER	
PRUEBA	BSET	PORTA,X,\$E0	
	BRA	REG	
NORMAL	BCLR	PORTA,X,\$C0	
	BSET	PORTA,X,\$20	
	BRA	LEER	
ALTO	BCLR	PORTA,X,\$C0	
	BSET	PORTA,X,\$A0	
	JSR	RETRASO	
	BCLR	PORTA,X,\$40	
	JSR	RETRASO	
	BRA	LEER	

Para programar el retardo observe:

FFFFh=65535d

Por 5 ciclos veces que se realiza el lazo, se

RETRASO	LDA	#\$Δ	(1)
LAZO	LDY	#\$FFFF	(2)
SUBLAZO	NOP		(1)
	DEY		(1)
	BNE	SUBLAZO	(3)
	DECA		(1)
	BNE	LAZO	(3)
	RTS		(5)

Realizan un LDY, un DECA y BNE, el cual suman 6 ciclos de instrucción.

$$4_{JSR} + 1_{LDA} + [6 + 5(65535)]\Delta + 5_{RTS}$$

ESTO DEBEMOS IGUALARLO A:

$$250\,000\text{ [Ms]} = \{4_{JSR} + 1_{LDA} + [6 + 5(65535)]\Delta + 5_{RTS}\}0.125\text{[Ms]}$$

$$\Delta = \{(250000/0.125) - 4 - 1 - 5\} / (6 + 5(65535))$$

Δ=6.10347 en decimal

POR LO TANTO:

$$\Delta = 6h$$