

### 1.12 CONVERTIDOR ANALÓGICO DIGITAL – UNA CONVERSIÓN CONFIGURACIÓN DE LECTURA DEL ADC

Preparado por: Rangel Alvarado

Estudiante Graduando de Lic. en Ing. Electromecánica

Universidad Tecnológica de Panamá

Panamá, Panamá

“e-mail”: [issaiass@cwpanama.net](mailto:issaiass@cwpanama.net)

“web site”: <http://www.geocities.com/issaiass/>

#### ÍNDICE

1.12.1	Introducción	166
1.12.2	Señal Analógica vs. Digital	167
1.12.3	El Convertidor Analógico Digital (ADC)	167
1.12.4	Registros del ADC	170
1.12.5	Diagrama de Flujo	171
1.12.6	Código	172
1.12.7	Simulador	175
1.12.8	Conclusión	175
1.12.9	Referencias	176
1.12.10	Problemas Propuestos	176

#### 1.12.1 Introducción

---

La mayoría de los sistemas digitales procesan señales booleanas (unos o ceros). Un cero (0) lógico está representado por el rango de 0 a 0.8 V y un uno (1) lógico está en el rango de los 2 a 5 V para el caso de las familias TTL. Cualquier sistema digital trabaja con solo las posibilidades uno (1) y cero (0); valores entre 0.8 y 2 V son indeterminados para un sistema digital.

En contraste, una cantidad analógica puede cambiar en cualquier rango de valores continuos y su valor exacto es significativo. Un ejemplo de variables analógicas son presión, temperatura, velocidad, entre algunas.

Un convertidor analógico digital o ADC transforma una cantidad eléctrica (por ejemplo, voltaje) en un valor que represente un código de salida digital.

En este abstracto se hace una descripción de cómo inicializar un ADC en el microcontrolador y aspectos importantes como:

- Señal Analógica vs. Señal Digital: se introduce la diferencia entre una señal real y una gráfica de captura de la señal en el mundo digital.
- El convertidor analógico Digital: para entender un ADC y su función en un sistema se necesitan entender de ciertas características importantes, como lo son la resolución y el tiempo de conversión.
- Registros del ADC: explica que registros existen en el microcontrolador y como inicializarlos.

Finalmente se explica un programa ejemplo de comprobación del buen funcionamiento del ADC y como se visualiza cada registro en el simulador.

## 1.12.2 Señal Analógica vs. Digital

Muchas de las señales que existen en el mundo real no son compatibles directamente con los pines de E/S del microcontrolador. Una señal analógica puede tomar cualquier valor sobre un rango continuo de valores, que no son exactamente un uno (1) lógico y un cero (0) lógico. Un ejemplo de señales analógicas pueden ser: temperatura, presión, señales de audio, velocidad de rotación, entre algunas.

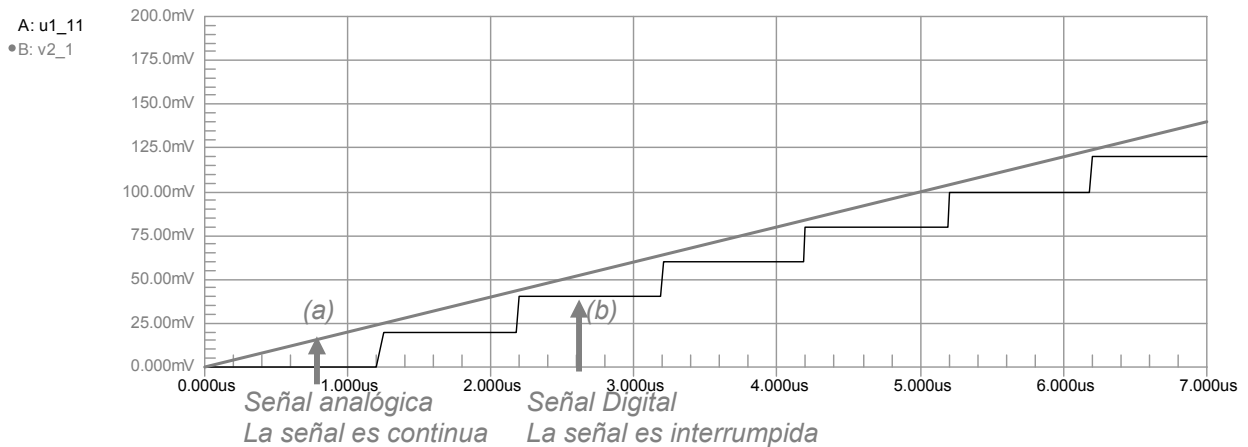


Figura 105. Señales Analógicas vs. Señales Digitales. (a) Señal analógica. Una señal analógica es continua en el tiempo. (b) Gráfico de captura de la señal análoga. Este es el posible gráfico de la captura de una señal analógica, pasada al mundo digital.

## 1.12.3 El Convertidor Analógico Digital (ADC)

Refiérase a la figura 106.

### 1.12.3.1 Operación del ADC en un Sistema

- (1) Por medio de transductores (sensores), la señal analógica es transformada a una cantidad eléctrica. Nota: Ciertas ocasiones, antes del paso (2) sigue un filtro RC.
- (2) Luego, el Convertidor Analógico a Digital (ADC) transforma esta cantidad eléctrica (generalmente en milivoltios) a un valor digital.
- (3) En seguida el convertidor pasa por la computadora, la cual procesa la información de acuerdo a las instrucciones del programador para generar alguna salida de control.

En sí la función de un ADC es tomar un voltaje de entrada analógico y después de cierto tiempo producir un código de salida digital que representa la entrada analógica.

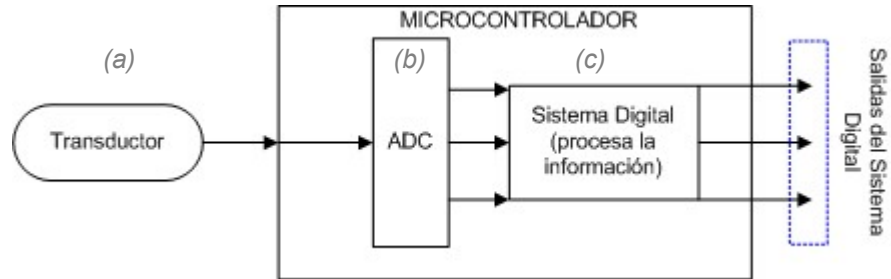


Figura 106. Sistema con ADC. (a) Transductor o sensor. Adquiere una señal analógica (p.e. temperatura) y la transforma a una variable eléctrica. (b) Convertidor Analógica Digital (ADC). Toma el voltaje analógico y lo transforma a su equivalente digital. (c) Sistema Digital. Procede a ejecutar instrucciones de control que genera una salida equivalente al programa.

**1.12.3.2 Características Importantes de un ADC**

**1.12.3.2.1 Resolución en un ADC**

La resolución es la cantidad más pequeña de cambio en la entrada analógica que produce un cambio de salida digital del ADC. Esta viene dada por la ecuación:

$$\text{Resolución} = \frac{V_{REFH}}{2^n}$$

Ecuación 2. Ecuación de Resolución de un ADC

$V_{REFH}$  = Voltaje de Referencia del convertidor.  
 n = Número de bits del convertidor

Ejemplo: Calcule la resolución en un ADC de 8 bits, con su voltaje de referencia a 5V.

$$\text{Resolución} = \frac{V_{REFH}}{2^n} = \frac{5}{2^8} = 19.531mV$$

La resolución depende del número de bits del convertidor. Cada 20mV de cambio, representa un aumento en el valor de salida del convertidor.

$$V_{OUT(analógico)} = \left( \frac{V_{REFH}}{2^n} \right) \times V_{OUT(digital)}$$

Ecuación 3. Voltaje de Salida Analógico

$V_{OUT(analógico)}$  = Voltaje de Salida análogo del sensor en voltios  
 $V_{OUT(digital)}$  = Voltaje de Salida digital (valor binario de 0 a 255)

Ejemplo: ¿Cuál será el resultado de un valor analógico de 2.23V en la salida del ADC?

$$V_{OUT(analógico)} = \left( \frac{V_{REFH}}{2^n} \right) \times V_{OUT(digital)}$$

$$V_{OUT(digital)} = \frac{V_{OUT(analógico)}}{\left( \frac{V_{REFH}}{2^n} \right)} = \frac{2.23}{\left( \frac{5}{2^8} \right)} = 114.18 = 114_{10} = 72_{16}$$

### 1.12.3.2.2 Tiempo de Conversión

Para este módulo ADC en especial, se necesitan 16 ciclos de reloj para, de este modo, conseguir una conversión. La ecuación siguiente aplica al ADC:

$$\text{Tiempo de Conversión} = \frac{16 \text{ ciclos de conversión}}{\text{Frecuencia del reloj del ADC}} = \frac{16}{XTAL / 2^{ADIV+2}}$$

*Ecuación 4. Tiempo de Conversión*

Tiempo de Conversión = Tiempo que toma el ADC en ejecutar una conversión  
 Frecuencia del reloj del ADC = Frecuencia del correspondiente bus del micro  
 XTAL = Cristal externo en Hz.  
 ADIV = Divisor dado por la tabla 48.

Ejemplo: Calcule el tiempo de conversión del convertidor, con un cristal de 4.9152 MHz, y un ADIV = 0 (ver tabla 48).

$$\text{Tiempo de Conversión} = \frac{16 \text{ ciclos de conversión}}{XTAL / 2^{ADIV+2}} = \frac{16}{4.9152 \times 10^6 / 4} = 13.03 \mu\text{seg}$$

En conclusión, la frecuencia de reloj del ADC es de 1.2288 MHz y la frecuencia de muestras (*1/tiempo de conversión*) capturadas es de 76.8 kHz.

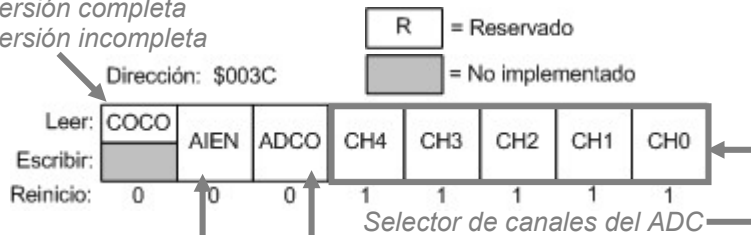
## 1.12.4 Registros del ADC

### 1.12.4.1 Registro de Estado y Control del ADC (ADSCR)

Nota: El bit COCO se borra al leer el registro.

1 = Conversión completa

0 = Conversión incompleta



1 = Interrupción Habilitada

0 = Interrupción Inhabilitada

1 = Conversión Continua

0 = Una sola conversión

Figura 107. Registro de Control y Estado del ADC (ADSCR). El bit COCO solo es uno (1) si la conversión se completa y AIEN = 0; si AIEN = 1, el bit COCO siempre será cero (0). El bit ADCO selecciona si se requiere una sola conversión o convertir seguidamente; mientras que los bits CH4 a CH0 seleccionan el puerto que actuará como ADC, ver tabla 47.

### 1.12.4.2 Registro de Datos del ADC (ADR)

El resultado de la conversión, luego de 16 ciclos se deposita en este registro

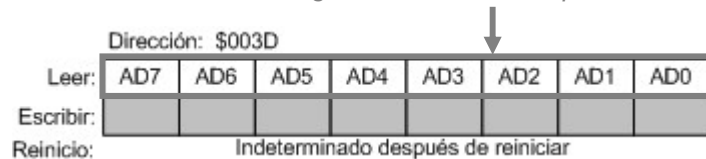


Figura 108. Registro de Datos del ADC (ADR). Un resultado digital de  $00_{16}$  a  $FF_{16}$  es almacenado en este registro, resultado de la lectura de un valor analógico.

Tabla 47. Selector de Canales del Convertidor Analógico Digital

CH4	CH3	CH2	CH1	CH0	Canal del ADC	Puerto
0	0	0	0	0	ADC0	PTB0
0	0	0	0	1	ADC1	PTB1
0	0	0	1	0	ADC2	PTB2
0	0	0	1	1	ADC3	PTB3
0	0	1	0	0	ADC4	PTB4
0	0	1	0	1	ADC5	PTB5
0	0	1	1	0	ADC6	PTB6
0	0	1	1	1	ADC7	PTB7
0	1	0	0	0	ADC8	PTD3
0	1	0	0	1	ADC9	PTD2
0	1	0	1	0	ADC10	PTD1
0	1	0	1	1	ADC11	PTD0
No utilizados o reservados (01100 hasta 11100)					-	-
1	1	1	0	1	-	V <sub>DDA</sub>
1	1	1	1	0	-	V <sub>SSA</sub>
1	1	1	1	1	-	ADC <sub>OFF</sub>

Nota: El valor por defecto del ADC es siempre estar deshabilitado, ver figura 107.

### 1.12.4.3 Registro de Reloj de Entrada del ADC (ADICLK)



La combinación de estos bits, cambia el divisor del reloj del ADC

Figura 109. Registro de Reloj de Entrada del ADC (ADICLK). Un resultado digital de  $00_{16}$  a  $FF_{16}$  es almacenado en este registro, resultado de la lectura de un valor analógico. Ver tabla 47 en donde se decide el puerto de conversión.

Tabla 48. Radio de División del Reloj del ADC

ADIV2	ADIV1	ADIV0	Rata del Reloj
0	0	0	ADICLK ÷ 1
0	0	1	ADICLK ÷ 2
0	1	0	ADICLK ÷ 4
0	1	1	ADICLK ÷ 8
1	X	X	ADICLK ÷ 16

X = No importa.

ADICLK = Reloj de entrada del ADC.

### 1.12.5 Diagrama de Flujo

El siguiente programa muestra como se puede inicializar el ADC y conectarlo al canal de referencia alto ( $V_{REFH}$ ) para verificar si está trabajando correctamente.

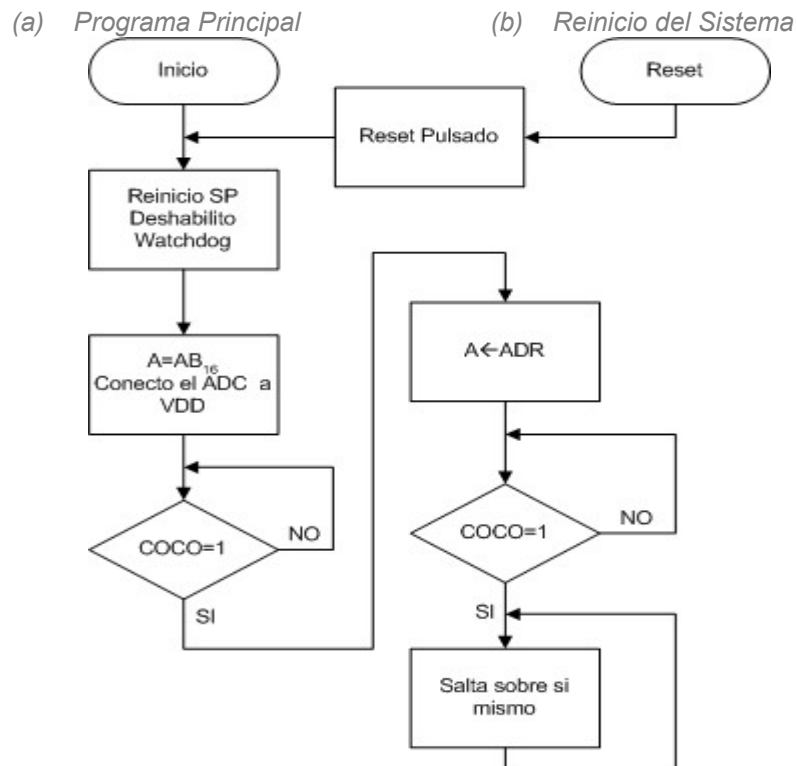


Figura 110. NT0012 – ADC – 1 Conversión. (a) Programa Principal. El programa configura el registro ADSCR para realizar una conversión, sin interrupciones y conectar el convertidor al voltaje de referencia para verificar la lectura de 5V ( $FF_{16}$ ), que corresponde a un buen funcionamiento del ADC. (b) Reinicio del Sistema. El sistema reinicia sin importar lo que haya estado haciendo el programa principal.

## 1.12.6 Código

```

=====
; ARCHIVO      : NT0012 - ADC - 1 Conversión - 08 03 04.asm
; PROPÓSITO   : PROGRAMA DEMOSTRATIVO PARA CORROBORAR EL BUEN
;              DESEMPEÑO DEL CONVERTIDOR ANÁLOGO DIGITAL
;              - Ver el resultado de captura del voltaje
;              de referencia del convertidor.
;              Solamente se muestra como configurar el conver
;              tidor análogo digital, y que sucede cuando se
;              configura para un tiempo de conversión y canal
;              conectado a Vss (GND de alimentación del micro)
;
;              P.D.: no hay que hacer ninguna conexión, esta
;              internamente cableado.
; Nota        :
;              1 - Ver el resultado de la dirección $003C cuando
;              completa la conversión (COCO = 1)
;              2 - Ver el resultado de la captura de la dirección
;              $003D (ADR = Rango de valor, en este caso #$FF)
;              3 - Ver el prescaler del ADC dependiendo del reloj
;              Oscilador externo [Xtal/2^(ADIV+2) ~ 1.23 MHz]
; REFERENCIA:
;              Advanced Information of MCU68HC908JK1, JK3, JL3...
;              http://www.freescale.com/files/microcontrollers/
;              doc/data_sheet/MC68HC08JL3.pdf
;              Pág. 33 - Registros del ADC
;              Pág. 35 - Vector de Interrupción
;              Págs. 127 @ 136. - Información del ADC
;
; LENGUAJE    : IN-LINE ASSEMBLER
=====
; HISTORIAL
; DD MM YY
; 10 02 03 Creado.
; 29 08 04 Modificado.
=====
; Pasos para iniciar el ADC :
;
; 1 - Definir del registro ADSCR o registro de
; control del ADC:
;     ¿Necesito o no interrupciones? (AIEN)
;     ¿Deseo una conversión o continua? (ADCO)
;     ¿Qué canal voy a usar? (CH[4:0])
; 2 - Definir el registro ADICLK o registro de reloj
; del ADC. Ajustarlo para un tiempo de conversión
; del ADC de 1MHz, aproximado... esto se logra:
;     ADC clock freq = XTAL/2^(ADIV+2)
=====

```

NT0012

Rev. 1 del 06.08.05

```

;=====
;
;                               Definiciones del Usuario
;=====
COPD      equ 0T                ; Bit 0 del registro CONFIG1
COCO      equ 7T                ; ADSCR, Bit de Conversión Completa, Bit 7,
; ON
CH4       equ %00010000        ; ADSCR, Bit de Canal 4, Bit 4 ON
CH3       equ %00001000        ; ADSCR, Bit de Canal 3, Bit 3 ON
CH2       equ %00000100        ; ADSCR, Bit de Canal 2, Bit 2 ON
CH1       equ %00000010        ; ADSCR, Bit de Canal 1, Bit 1 ON
CH0       equ %00000001        ; ADSCR, Bit de Canal 0, Bit 0 ON
VDDA      equ CH4|CH3|CH2|CH0; ADSCR, Grupo de Bits, Bit 4, 3, 2, 0 ON
ADIV      equ %00000000        ; ADIV = 0, ADICLK

;=====
;
;                               Registro de Configuraciones
;=====
CONFIG1   equ $001F            ; Vectores de configuración

;=====
;
;                               Registro ADC
;=====
ADSCR     equ $003C            ; Dirección, registro de estado y control del
; ADC
ADR       equ $003D            ; Registro de Datos del ADC
ADICLK    equ $003E            ; Registro de Reloj de Entrada del ADC

;=====
;
;                               Memoria FLASH
;=====
FLASH_START equ $EC00          ; Puntero - Mem. FLASH

;=====
;
;                               Vectores de Usuario
;=====
RESET_VEC equ $FFFE            ; Puntero del RESET

;=====
; OBJETIVO : Inicio de Codif. del Ensam-
;           blador en Memoria FLASH.
;=====
org FLASH_START                ; Inicio Mem. FLASH

```



```

;=====
; OBJETIVO   : Observar el resultado de una
;              conversión completa del ADC en VREFH
;=====
START
    rsp                ; Inic.Stack = $00ff
    bset COPD,CONFIG1 ; Desactiva watchdog
    lda #$AB           ; Carga A con #$AB, para que se vea el
                        ; cambio a cero en el convertidor.
    mov #ADIV,ADICLK  ; Prescaler del ADC de 0 = #%00000000
                        ; hay una frecuencia de muestreo de
                        ; aproximadamente 76.8khz
    mov #VDDA,ADSCR   ; ADCO = 0 CH1 = 0, asegura que
                        ; #%0001 1101 se de solo una conversión y se
                        ; muestre el resultado de Vdd
    brclr COCO,ADSCR,* ; Espera a terminar la conversión
                        ; ayuda a estabilizar el convertidor
                        ; En teoría, al completar la
                        ; Conversión, si el Bit7 se pone
                        ; a 1 ADSCR debe de ser 9E
                        ; Debido a que el bit COCO ya
                        ; anunció una conversión completa.
    lda ADR            ; Muestra el resultado en el ACCA
                        ; En teoría deberá cargar el ACCA
                        ; con el Valor de #$FF
    brclr COCO,ADSCR,* ; Espera a terminar la conversión
                        ; Nunca termina porque solo se seleccionó
                        ; una sola conversión
    bra *              ; Salta sobre si mismo

;=====
; OBJETIVO   : Inicializa el Vector de Reset
;              Arranque del programa en la
;              memoria Flash.
;=====
;==== Vector de Reinicio de Sistema =====
    org RESET_VEC     ; Puntero Vec - RESET
    dw START           ; al darse reset salta a Start

```

*Listado 10. NT0012 – ADC – Una Conversión. La función principal del programa es demostrar como configurar el convertidor analógico digital para observar valores analógicos en forma digital, en este caso se muestra un código de diagnóstico para el programa y demostrar que el ADC opera perfectamente, registrando los voltajes de referencia, en este caso el voltaje de referencia alto (5V).*

## 1.12.7 Simulador

En esta sección, refiérase a la figura 111.

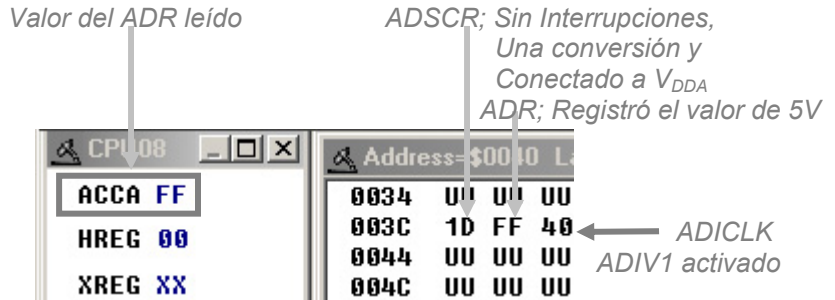


Figura 111. Registros del ADC en el Simulador. El registro de control fue configurado para el ADC en una conversión, sin interrupciones y se conectó al canal que comunica a  $V_{DDA}$ ; el reloj interno del ADC está configurado para XTAL/4, activando el bit ADIV1. Luego de ciertos ciclos de máquina el registro ADR, retiene el valor de la conversión.

- Inicie WinIDE.
- Seleccione el archivo NT0012 – ADC – 1 Conv 08 03 04.asm
- Compile.
- Inicie el Simulador en el modo de simulación en Circuito (“Contact target with this settings...”)
- Corra el programa paso a paso (botón “step”)
- Verifique los registros ACCA, ADSCR, ADICLK, ADR y obtenga sus propias conclusiones.

## 1.12.8 Conclusión

El convertidor analógico digital se utiliza para interpretar cantidades analógicas de sensores y capturar su resultado de manera digital para luego procesarlo digitalmente, esto permite tener control de variables físicas como temperatura, presión e intensidad.

Con el fin de explicar que es un ADC, se diferencié una señal analógica de una digital; como opera un ADC en un sistema, sus características más importantes, registros en el microcontrolador JK3 y JL3 además de una prueba en el simulador que vincula donde se ubica cada registro.

## 1.12.9 Referencias

---

### 1.12.9.1 Sistemas Digitales, Principios y Aplicaciones

Autor: Ronald J. Tocci

Recurso: Capítulo 10 – Interfase con el Mundo Analógico

### 1.12.9.2 Información Avanzada sobre el Microcontrolador

(a) [http://www.freescale.com/files/microcontrollers/doc/data\\_sheet/MC68HC08JL3.pdf](http://www.freescale.com/files/microcontrollers/doc/data_sheet/MC68HC08JL3.pdf)

*Pág. 33 - Registros del ADC*

*Pág. 35 - Vector de Interrupción*

*Págs. 127 – 136. - Información del ADC*

### 1.12.9.3 Manual de Referencia del CPU

(a) [http://www.freescale.com/files/microcontrollers/doc/ref\\_manual/CPU08RM.pdf](http://www.freescale.com/files/microcontrollers/doc/ref_manual/CPU08RM.pdf)

*Pág. 155, Instrucción MOV.*

### 1.12.9.4 Página “web” sobre esta Nota Técnica

(a) <http://www.geocities.com/issaiass/>

## 1.12.10 Problemas Propuestos

---

1.12.10.1 Cambie el código utilizado para generar un reloj de entrada del ADC de 1/16 del valor del cristal.

1.12.10.2 Cambie el código utilizado para conectarse al canal de conversión de referencia bajo ( $V_{SSA}$ ) y registre su resultado en el acumulador (ACCA).