

## ADC: Convertidor Analógico/Digital (section 16)

La mayoría de los equipos electrónicos actuales son digitales, pero como bien sabemos, el mundo real es analógico. Podría decirse que la misión de los convertidores (tanto analógico/digitales ADC = *Analog-to-Digital Converter*, como digital/analógicos, DAC = *Digital-to-Analog Converter*) es unir ambos mundos.

Para no perder información, dicha conversión debe ser lo mejor posible, ya que el objetivo de un convertidor A/D es que el valor digital que proporciona a su salida sea un reflejo, lo más preciso posible, del valor analógico que se le proporciona como entrada.

A mayor número de bits en la salida digital, mayor será la precisión, pero también aumentará la complejidad del circuito, su precio etc. De hecho, aumentar en un bit el número de bits de la salida supone duplicar la precisión (y la resolución), pero junto con ello se complica el diseño del circuito, resultando generalmente más lento. Así pues, es necesario un compromiso entre esos dos factores, y hay que elegir el convertidor que mejor se adapte a las necesidades de las aplicaciones que se quieran implementar.

La figura 1 muestra la estructura de un pin analógico:

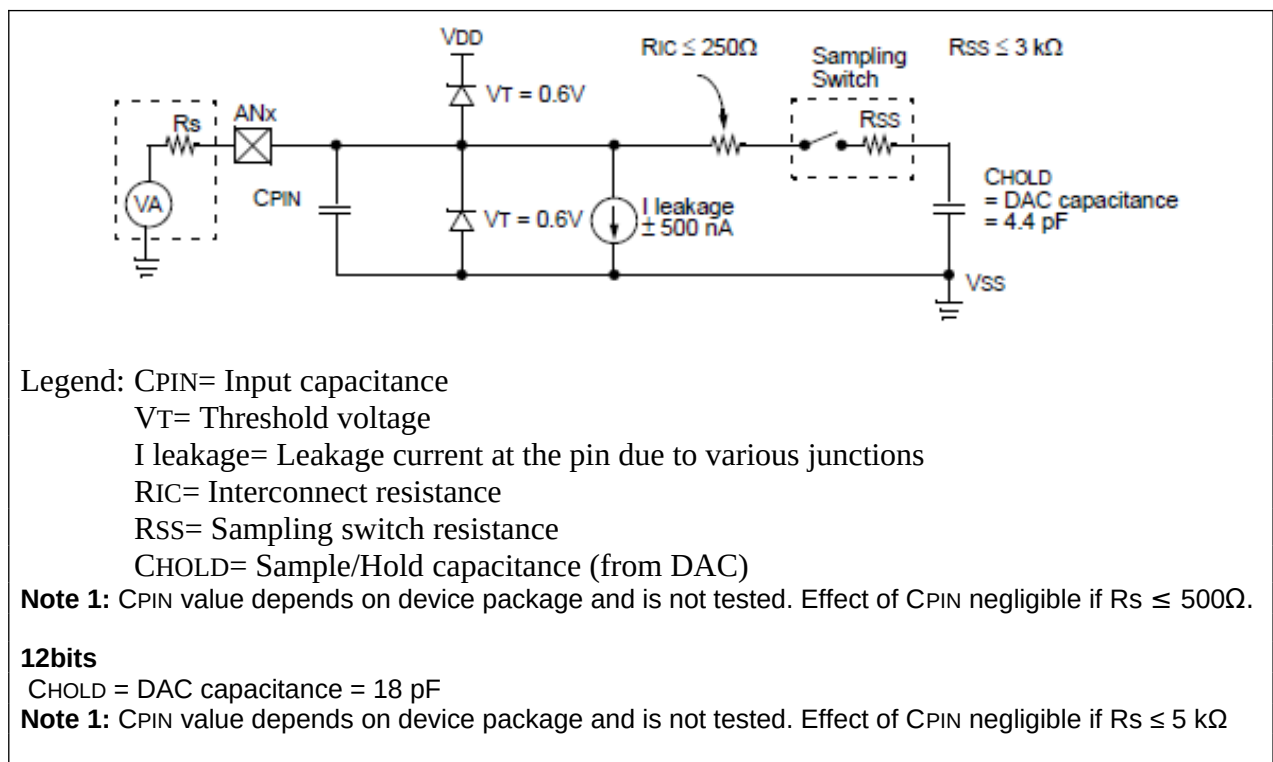


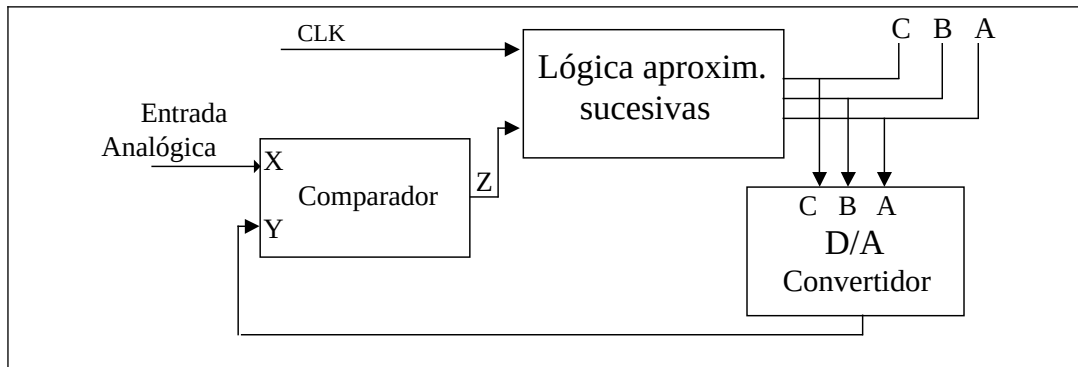
Figura 1. Estructura de un pin analógico.

Un convertidor A/D se compone de dos circuitos o partes:

- El “muestreador” (en inglés *Sample & Hold (S/H)* o *Track & Hold (T/H)*), que carga y mantiene constante el valor de la señal analógica recogida mientras dure la conversión.
- El cuantificador digital, que realiza la conversión numérica y obtiene el valor digital de salida.

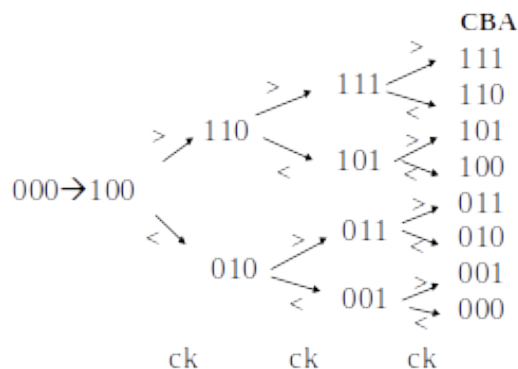
Uno de los parámetros más representativos de un convertidor A/D es el número de bits de salida ‘n’ que tiene, lo que significa que el convertidor identifica  $2^n$  niveles de tensión diferentes. Es decir, cada valor de salida diferente representa una fracción de la tensión de referencia (rango de tensiones entre los que funciona). Por ejemplo, si estamos trabajando con un rango de tensión de 5 voltios, con  $n=8$ , tendremos  $2^8=256$  niveles posibles (0-255), y cada nivel indica una variación de 0,0195 V (19,5 mV).

Los convertidores A/D del PIC24H utilizan el denominado “método de aproximaciones sucesivas” para hacer la cuantificación, siendo su estructura la que se muestra a continuación:



En el método de aproximaciones sucesivas la señal analógica muestreada se compara con diferentes valores digitales, decidiendo en cada comparación el valor de uno de los bits de salida. Se comienza con el bit de mayor peso y se sigue en orden descendente hasta obtener el código digital de salida. El tiempo necesario para obtener dicho código depende de la señal de reloj con la que trabaja este módulo, ya que en cada ciclo de reloj se calcula el valor de uno de los bits de salida.

Por ejemplo, si la salida digital es de 3 bits, la primera comparación se haría con el valor digital **100**. El convertidor D/A interno obtiene la señal analógica correspondiente a ese valor y se compara con la señal de entrada. Si la entrada es mayor, se fijará como valor del bit de más peso 1, y si no, se fijará 0. Así pues, el siguiente valor con el que comparar será el **110** o el **010** según el caso, y siguiendo la misma lógica se establecerá el valor de ese segundo bit. El proceso se repite tantas veces como bits de salida tengamos, y el número de ciclos necesarios es igual al número de bits de la salida.



## Convertidor A/D en el PIC24H

El PIC24H cuenta con dos módulos convertidores A/D y cada uno de ellos se puede configurar para que la salida digital sea de 10 o de 12 bits. Estas son algunas de las características de los convertidores ADC:

- Velocidad de conversión: hasta 1,1MSps (*Mega Sample Per Second*). Con 12 bits, 500 kSps.
- Hasta 32 canales analógicos (AN0 – AN31). El módulo ADC2 sólo 16 (AN0-AN15).
- Se pueden escanear hasta 4 salidas al mismo tiempo. Con 12 bits, solo una entrada.
- El modo de escanear o analizar la entrada es automático pero el modo de disparo del convertidor es programable.
- No tiene buffer para guardar los datos, pero puede utilizar DMA.
- Cuenta con 4 formas diferentes de alinear los resultados.

La figura 2 muestra el diagrama de bloques del convertidor A/D en el PIC24H.

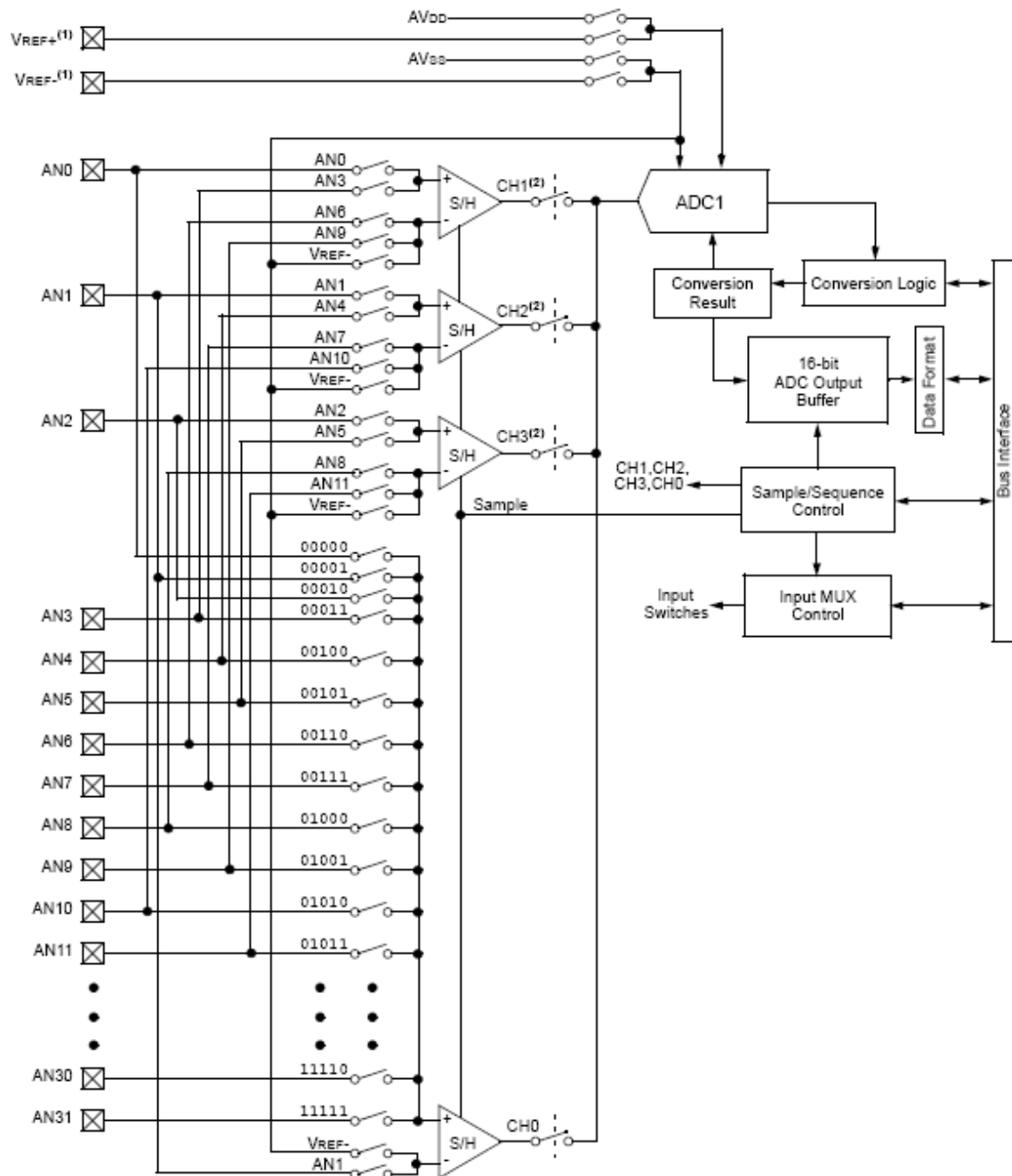
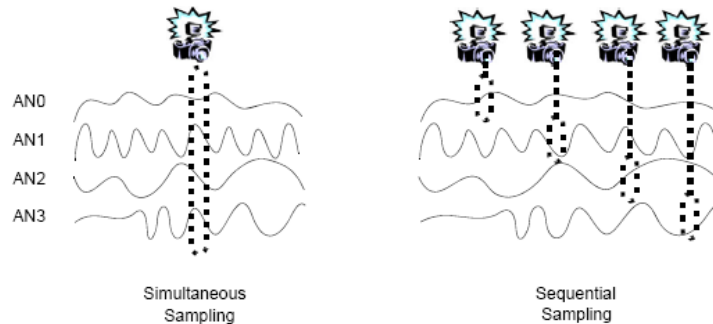


Figura 2. Diagrama del convertidor A/D del PIC24H

Como ya se ha indicado anteriormente, es posible utilizar el DMA (*Direct Memory Access*) y almacenar directamente los datos obtenidos por el convertidor en memoria RAM sin mediar la CPU. Para ello se utiliza un buffer en el espacio de memoria asignado al DMA.

Durante el proceso de conversión se pueden alternar las entradas analógicas, asociadas en dos grupos. Además, las entradas analógicas está multiplexadas en cuatro canales (*Sample & Hold*) (CH0-CH3) permitiendo 1, 2 ó 4 adquisiciones de datos simultáneas.



El convertidor A/D de 10 bits puede usar 4 canales S&H: CH0, CH1, CH2 y CH3. Los S&H están conectados a las patas analógicas mediante un multiplexor analógico que se controla desde los registros ADxCHS123 y ADxCHS0. En este registro existen dos grupos de bits para el control del multiplexor que funcionan del mismo modo. Estos bits permiten programar dos configuraciones distintas del multiplexor analógico de entrada que se denominan *Sample A* (MUX A) y *Sample B* (MUX B). El convertidor A/D puede opcionalmente permutar de configuración *Sample A* (MUX A), a configuración *Sample B* (MUX B) y viceversa.

El número de circuitos S&H que se utilizan en la secuencia de muestreo y de conversión, queda determinado por los bits de control CHPS<1:0> (registro ADxCON2<9:8>). El bit SIMSAM (ADCON1<3>), determina si el muestreo se hace de forma simultánea o secuencial. Un muestreo simultáneo (SIMSAM= 1) se realiza en el mismo instante de tiempo para todas y cada una de las entradas analógicas. Por otro lado, en el muestreo secuencial el instante de muestreo para cada una de las entradas analógicas es diferente.

La figura 3 muestra con mayor detalle los esquemas de los 4 canales con los que cuenta el convertidor.

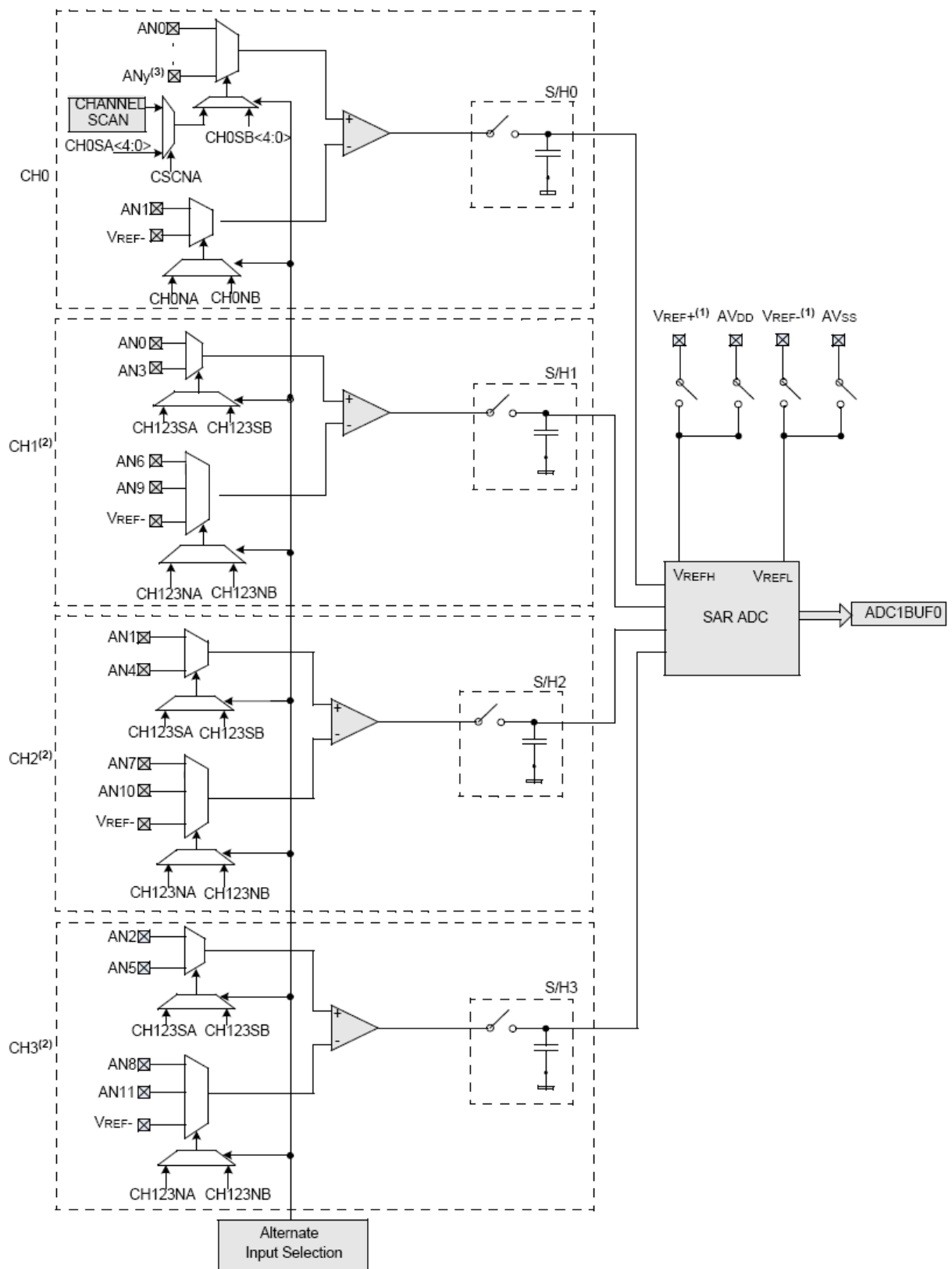


Figura 3. Detalle de los canales S&H con los que cuenta el convertidor A/D.

El módulo ADC1 que vamos a utilizar tiene 10 registros de control y/o estado:

- AD1CON1: *ADCx Control Register 1*
- AD1CON2: *ADCx Control Register 2*
- AD1CON3: *ADCx Control Register 3*
- AD1CON4: *ADCx Control Register 4*
- AD1CHS0: *ADCx Input Channel 0 Select Register*
- AD1CHS123: *ADCx Input Channel 1, 2, 3 Select Register*
- AD1PCFGH: *ADC1 Port Configuration Register High*
- AD1PCFGL: *ADCx Port Configuration Register Low*
- AD1CSSH: *ADC1 Input Scan Select Register High*
- AD1CSSL: *ADCx Input Scan Select Register Low*

Los registros AD1CONn son los encargados de controlar la operación del convertidor. Los registros AD1CHS0 y AD1CHS123 seleccionan los pines de entrada que se conectan a los amplificadores de muestreo y retención (S&H). Los registros AD1PCFGH y AD1PCFGL se encargan de configurar los pines como analógicos (entradas) o como digitales (entradas o salidas). Finalmente, AD1CSSH y AD1CSSL seleccionan las entradas que van a ser exploradas secuencialmente.

## AD1CON1: A / D Control Register 1

R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0, HCS	R/W-0, HCS
ADON	-	ADSIDL	ADDMABM	—	AD12B	FORM<1:0>	SSRC<2:0>	-	SIMSAM	ASAM	SAMP	DONE
Bit15						Bit9-8	Bit7-5					bit0

Legend: R= readable bit W= writable bit HC=Cleared by Hardware HS= Set by Hardware

**ADON:** Habilita el CAD.

*1 = El módulo A/D está operativo. / 0 = apagado*

**ADSIDL:** Permite el funcionamiento en modo *IDLE*.

*1 = Módulo detenido al entrar en modo IDLE / 0 = módulo continúa funcionando en IDLE*

**ADDMABM:** Modo del buffer de DMA

*1 = El buffer del DMA se escribe en el orden de la conversión*

*0 = El buffer del DMA se escribe en modo Scatter/Gather*

**AD12B:** Selecciona el modo de operación de 10 ó 12 bits

*1 = 12 bits, 1 canal      0 = 10 bits, 4 canales*

**FORM <1:0>:** Selecciona el formato de los datos de salida (*11, 10 no disponibles en el PIC24HJ256GP610A*)

d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

*11 = Signed fractional (sddd dddd dd00 0000) (No todos los tienen)*

d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	---	---	---	---	---

*10 = Fractional (dddd dddd dd00 0000) (No todos los tienen)*

d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	---	---	---	---	---

*01 = Signed integer (ssss sssd dddd dddd)*

d09	d09	d09	d09	d09	d09	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

*00 = Integer (0000 00dd dddd dddd)*

0	0	0	0	0	0	0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
---	---	---	---	---	---	---	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

V <sub>IN</sub> /V <sub>REF</sub>	10-Bit Output Code	16-Bit Integer Format/ Equivalent Decimal Value		16-Bit Signed Integer Format/ Equivalent Decimal Value	
1023/1024	11 1111 1111	0000 0011 1111 1111	1023	0000 0001 1111 1111	511
1022/1024	11 1111 1110	0000 0011 1111 1110	1022	0000 0001 1111 1110	510
***					
513/1024	10 0000 0001	0000 0010 0000 0001	513	0000 0000 0000 0001	1
512/1024	10 0000 0000	0000 0010 0000 0000	512	0000 0000 0000 0000	0
511/1024	01 1111 1111	0000 0001 1111 1111	511	1111 1111 1111 1111	-1
***					
1/1024	00 0000 0001	0000 0000 0000 0001	1	1111 1110 0000 0001	-511
0/1024	00 0000 0000	0000 0000 0000 0000	0	1111 1110 0000 0000	-512

V <sub>IN</sub> /V <sub>REF</sub>	10-Bit Output Code	16-Bit Fractional Format/ Equivalent Decimal Value		16-Bit Signed Fractional Format/ Equivalent Decimal Value	
1023/1024	11 1111 1111	1111 1111 1100 0000	0.999	0111 1111 1100 0000	0.499
1022/1024	11 1111 1110	1111 1111 1000 0000	0.998	0111 1111 1000 0000	0.498
***					
513/1024	10 0000 0001	1000 0000 0100 0000	0.501	0000 0000 0100 0000	0.001
512/1024	10 0000 0000	1000 0000 0000 0000	0.500	0000 0000 0000 0000	0.000
511/1024	01 1111 1111	0111 1111 1100 0000	0.499	1111 1111 1100 0000	-0.001
***					
1/1024	00 0000 0001	0000 0000 0100 0000	0.001	1000 0000 0100 0000	-0.499
0/1024	00 0000 0000	0000 0000 0000 0000	0.000	1000 0000 0000 0000	-0.500

**SSRC<2:0>**: Selecciona la fuente de disparo de la digitalización.

111 = El contador interno finaliza el muestreo y comienza la digitalización (conversión automática)

110, 101, 100, 011= Reservado

010 = Timer3 para ADC1 y Timer5 para ADC2: termina el muestreo y comienza la conversión

001 = Un flanco ascendente en el pin INT0 finaliza el muestreo y comienza la conversión.

000 = Poniendo a '0' el bit SAMP finaliza la captura de datos y comienza la conversión.

**SIMSAM**: Selecciona muestreo simultáneo (sólo aplicable cuando CHPS = 01 o 1x).

1 y CHPS = 1x Muestrea simultáneamente CH0, CH1, CH2, CH3,

1 y CHPS = 01 Muestrea simultáneamente CH0 y CH1

0 = Muestrea múltiples canales individualmente de forma secuencial.

**ASAM**: Bit de auto-inicio de muestreo del CAD.

1 = El muestreo comienza inmediatamente después de finalizar la última conversión. El bit SAMP se desactiva automáticamente.

0 = El muestreo se inicia cuando el bit SAMP se pone a uno.

**SAMP**: Bit de habilitación del muestreo.

1 = Al menos uno de los S&H está funcionando

0 = Ningún S&H está funcionando

Con ASAM = 0, el muestreo comienza cuando si se escribe un '1' en este bit.

Si SSRC = 000, escribiendo un '0' finaliza el muestreo y comienza la conversión.

Si SSRC ≠ 000, el hardware lo borra automáticamente al finalizar el muestreo y comenzar la conversión

**DONE**: Bit de estado de la digitalización.

1 = Digitalización completada. / 0 = digitalización no completada o no ha comenzado

Pasa a '1' automáticamente al final de la digitalización. Se puede poner a '0' por software o automáticamente al comenzar una nueva conversión. El cambio de este bit durante el funcionamiento no afecta a las conversiones en proceso.

## AD1CON2: A / D Control Register 2

R/W-0	U-0	U-0	R/W-0	R/W-0	R-0	U-0	R/W-0	R/W-0	R/W-0
VCFG<2:0>	-	-	CSCNA	CHPS<1:0>	BUFS	-	SMPI<3:0>	BUFM	ALTS
Bit15				Bit8	Bit7				bit0

**VCFG<2:0>:** Configura la tensión de referencia

	V <sub>REFH</sub>	V <sub>REFL</sub>
000	AV <sub>DD</sub>	AV <sub>SS</sub>
001	External V <sub>REF+</sub>	AV <sub>SS</sub>
010	AV <sub>DD</sub>	External V <sub>REF-</sub>
011	External V <sub>REF+</sub>	External V <sub>REF-</sub>
1xx	AV <sub>DD</sub>	AV <sub>SS</sub>

**CSCNA:** Selecciona escanear entradas

1 = Entradas escaneadas por CH0+ durante Sample A

0 = Entradas no escaneadas

**CHPS<1:0>:** Selecciona los canales utilizados en la conversión.

Si AD12B= 1 se lee como '0' (CHPS<1:0> es U-0)

Si AD12B= 0:

1x = Canales CH0, CH1, CH2 y CH3.

01 = Canales CH0 y CH1

00 = Canal CH0

**BUFS:** (solo de lectura) Indica el estado de llenado del buffer. Sólo es válido cuando BUFM = 1

1 = Se está llenando segunda mitad del buffer. El usuario debe acceder a la primera mitad

0 = Se está llenando primera mitad del buffer. El usuario debe acceder a la segunda mitad

**SMPI<3:0>:** Incrementan el DMA o generan una interrupción con un número concreto de muestreos y digitalizaciones.

Con DMA:

1111 = Incrementa dir. DMA después de 16 secuencias de muestreo y conversión

1110 = Incrementa dir. DMA después de 15 secuencias de muestreo y conversión

...

0001 = Incrementa dir. DMA después de 2 secuencias de muestreo y conversión

0000 = Incrementa dir. DMA después de 1 secuencias de muestreo y conversión

**Sin DMA:** en vez de incrementar la dirección del DMA, genera una interrupción ADC

**BUFM:** Selecciona el modo de rellenar el buffer.

1 = la primera interrupción se genera al llenarse la primera mitad del buffer y la siguiente interrupción se genera al llenarse la segunda mitad del buffer

0 = Siempre se empieza a llenar el buffer desde la primera dirección

**ALTS:** Selección de modo de muestreo de entrada alternativa.

1 = Utiliza el canal seleccionado por Sample A para el 1er muestreo, y Sample B para el 2º

0 = Utiliza en todo momento los canales de entrada seleccionados para Sample A



## AD1CON3: A / D Control Register 3

R/W-0	U-0	U-0	R/W-0					R/W-0				
ADRC	—	—	SAMC<4:0>					ADCS7				
Bit15			Bit8					Bit7				
												bit0

**ADRC:** Indica la fuente del reloj del CAD

1 = Reloj RC interno, del módulo A/D.

0 = Reloj obtenido de la señal de reloj del sistema.

**SAMC<4:0>:** Selecciona el tiempo del muestreo automático.

11111 = 31 TAD

...

00001 = 1 TAD

00000 = 0 TAD

**ADCS<7:0>:** Selecciona el periodo del reloj de conversión ( $T_{AD}$ ) .

11111111 = Reservado

...

01000000 = Reservado

00111111 =  $T_{cy} (ADCS<7:0> + 1) = 64 T_{cy} = T_{AD}$

...

00000001 =  $T_{cy} (ADCS<7:0> + 1) = 2T_{cy} = T_{AD}$

00000000 =  $T_{cy} (ADCS<7:0> + 1) = T_{cy} = T_{AD}$

## AD1CON4: A / D Control Register 4

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
-	-	-	-	-	-	-	-	-	-	-	-	-	DMABL<2:0>
Bit15							Bit8	Bit7					bit0

**DMABL<2:0>:** Selecciona número de palabras reservadas en DMA por entrada analógica

111 = Reserva 128 palabras del buffer para cada entrada analógica

110 = Reserva 64 palabras

101 = Reserva 32

100 = Reserva 16

011 = Reserva 8

010 = Reserva 4

001 = Reserva 2

000 = Reserva 1 palabra del buffer para cada entrada analógica

## AD1CHS123: A / D Input Channel 1, 2, 3 Select Register

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
-	-	-	-	-	CH123NB<1:0>	CH123SB	-	-	-	-	-	-	CH123NA<1:0>	CH123SA	
Bit15					Bit8		Bit7					bit0			

Legend: R= readable bit W= writable bit HC=Cleared by Hardware HS= Set by Hardware

El registro **ADCHS** selecciona los pines de entrada que se van a conectar al CAD.

**CH123NB<1:0>**: Selección de entrada negativa de los canales 1, 2, 3 para Sample B

**CH123NA<1:0>**: Selección de entrada negativa de los canales 1, 2, 3 para Sample A

*Cuando AD12B =1, CHxNB es: U-0 (Unimplemented, se lee como '0')*

11 = AN9 es la entrada negativa de CH1

AN10 es la entrada negativa de CH2

AN11 es la entrada negativa de CH3

10 = AN6 es la entrada negativa de CH1

AN7 es la entrada negativa de CH2

AN8 es la entrada negativa de CH3

0x = V<sub>REFL</sub> es la entrada negativa de CH1, CH2, CH3

**CH123SB**: Selección de entrada positiva de los canales 1, 2, 3 Sample B

**CH123SA**: Selección de entrada positiva de los canales 1, 2, 3 Sample A

1 = AN3 es la entrada positiva de CH1

AN4 es la entrada positiva de CH2

AN5 es la entrada positiva de CH3

0 = AN0 es la entrada positiva de CH1

AN1 es la entrada positiva de CH2

AN2 es la entrada positiva de CH3

## AD1CHS0: A / D Input Channel 0 Select Register

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NB	—	—	CH0SB<4:0>	CH0NA	—	—	CH0SA<4:0>								
Bit15				Bit8			Bit7			bit0					

**CH0NA**: Selección de entrada negativa del canal 0 para Sample A

**CH0NB**: Selección de entrada negativa del canal 0 para Sample B

1 = AN1 es la entrada negativa de CH0

0 = V<sub>REFL</sub> es la entrada negativa de CH0

**CH0SB<4:0>**: Selección de entrada positiva del canal 0 para Sample B.

**CH0SA<4:0>**: Selección de entrada positiva del canal 0 para Sample A.

11111 = AN31 es la entrada positiva de CH0

11110 = AN30 es la entrada positiva de CH0

...

00001 = AN1 es la entrada positiva de CH0

00000 = AN0 es la entrada positiva de CH0

## AD1CSSH: A / D Input Scan Select Register for MUX A(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSSL31	CSSL30	CSSL29	CSSL28	CSSL27	CSSL26	CSSL25	CSSL24	CSSL23	CSSL22	CSSL21	CSSL20	CSSL19	CSSL18	CSSL17	CSSL16
Bit15				Bit8				Bit7				bit0			

## AD1CSSL: A / D Input Scan Select Register for MUX A(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSSL15	CSSL14	CSSL13	CSSL12	CSSL11	CSSL10	CSSL9	CSSL8	CSSL7	CSSL6	CSSL5	CSSL4	CSSL3	CSSL2	CSSL1	CSSL0
Bit15				Bit8				Bit7				bit0			

Los registros AD1CSSH y ADxCSSL seleccionan el orden en el que las entradas son escaneadas de forma secuencial.

**CSS<31:16> y CSS<15:0>**: Son los bits de selección de los pines de entrada que se escanean:

1 = Selecciona el canal ANx, para ser escaneado.

0 = Canal ANx NO SELECCIONADO

Nota: el CAD 2 solo tiene 16 canales, por lo que solo tiene CSS<15:0>

## AD1PCFGH: A / D Port Configuration Register High

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG31	PCFG30	PCFG29	PCFG28	PCFG27	PCFG26	PCFG25	PCFG24	PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18	PCFG17	PCFG16
Bit15				Bit8				Bit7				bit0			

## AD1PCFGL: A / D Port Configuration Register Low

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0
Bit15				Bit8				Bit7				bit0			

Los registros AD1PCFGH y ADxPCFGL configuran los pines ANx como analógicos (PCFGx= 0) o digitales (PCFGx= 1). Si son analógicos sólo pueden ser entrada, y si son digitales pueden ser entrada o salida. Cuando el micro se resetea, estos registros se ponen a '0', es decir, que los pines ANx se configuran como entradas analógicas, por lo que hay que revisar nuestras necesidades y asignarles el valor del bit adecuado según lo que vayamos a utilizar. Los TRIS de los pines analógicos deben programarse de ENTRADA.

## RESUMEN

Tal y como se ha comentado, una conversión analógico-digital es un proceso que consta de dos pasos: muestreo y digitalización. En el primer paso se toma una muestra de la tensión de la señal de entrada y, a continuación, se desconecta la entrada. En el segundo paso se asigna un valor binaria a la muestra. Las dos fases distintas son controladas por dos bits del registro de control AD1CON1: SAMP y DONE. Es importante respetar el tiempo de las dos fases para conseguir la precisión necesaria en todo el proceso.

Duante la **fase de muestreo** la señal externa se conecta a un condensador interno (*sample & hold*) que se cargará a la tensión de entrada siempre y cuando la conexión dure lo suficiente. El tiempo de carga que se necesita es proporcional a la impedancia de la señal de entrada y al valor interno del condensador.

El muestreo puede iniciarse mediante programa (poniendo a '1' el bit SAMP) o automáticamente (mediante el hardware del propio módulo ADC). Del mismo modo, puede finalizarse por programa (poniendo a '0' el bit SAMP) o automáticamente (sincronizándose con la señal de reloj del módulo ADC). Si SSRC <2:0> = 7 (111 en binario), el fin del muestreo y comienzo de la digitalización se fija mediante el reloj del ADC. El tiempo de muestreo automático viene determinado por el campo SAMC<4:0> (en el registro AD1CON3) que determina cuántos ciclos

$T_{AD}$  transcurren entre el comienzo del muestreo y el comienzo de la digitalización. Este modo de funcionamiento consigue velocidades de conversión altas en varios canales.

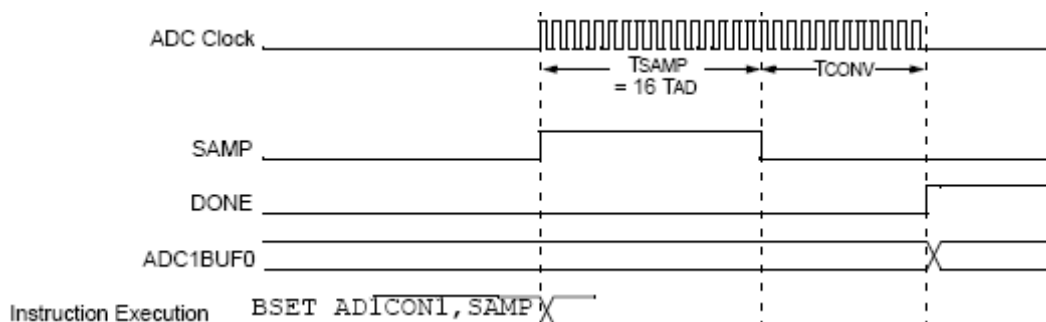
En el ejemplo 1 se puede ver que el muestreo comienza por programa (SAMP=1); dado que SSRC=7, el muestreo finaliza automáticamente pasado cierto tiempo (el fijado por el campo SAMC) y comienza la digitalización. Cuando la digitalización termine, el bit DONE se activará automáticamente.

## EJEMPLO 1

```
AD1PCFGL = 0xEFFF;           // Todos los PORTB digitales, excepto RB12 = analógico
AD1CON1 = 0x00E0;           // SSRC bit = 111 auto-convert
AD1CHS0 = 0x000C;           // RB12/AN12 a la entrada de CH0
AD1CSSL = 0;
AD1CON3 = 0x1F02;           // Tiempo muestreo = 31Tad, Tad = 2 Tcy
AD1CON2 = 0;                // tensión de referencia
AD1CON1bits.ADON = 1;       // ctivar módulo ADC1

while (1)
{
    AD1CON1bits.SAMP = 1;     // Comienza muestreo y 31Tad después la digitalización
    while (!AD1CON1bits.DONE); // Espera a que termine la digitalización (encuesta)
    ADCValue = ADC1BUF0;      // Deja el valor en la variable ADCValue y repite el proceso
}
```

Los bits SAMP (ADCON1<1>) y DONE (ADCON1<0>) indican el estado de muestreo y de la digitalización del módulo A/D, respectivamente. Generalmente, cuando el *bit* SAMP está a '0', indica el final del muestreo, mientras que cuando el *bit* DONE toma el valor '1' indica el final de la conversión (muestreo y digitalización).



La fase de **digitalización** depende de la señal de reloj del ADC, es decir, de  $T_{AD}$ . Dicha señal de reloj puede tener distintas fuentes: se puede obtener a partir del reloj de la CPU mediante un divisor de frecuencia, pero también de un oscilador interno de tipo RD independiente. La segunda opción es más adecuada en caso de que se quiera realizar una conversión estando la CPU en modo sleep (parada). Pero lo más habitual es utilizar el divisor de frecuencia.

Es conveniente configurar la señal de reloj del ADC para que respetando las restricciones del módulo, sea lo más rápida posible (menor periodo). En nuestro caso, el manual del micro indica que  $T_{AD} > 75 \text{ ns}$ , y para calcular dicho periodo se utilizan las siguientes fórmulas:

$$T_{AD} = T_{CY} [ADCS\langle 7:0 \rangle + 1]$$

$$ADCS\langle 7:0 \rangle = (T_{AD} / T_{CY}) - 1$$

Tal y como expresan las fórmulas, el campo ADCS (8 bits, en el registro ADxCON3) establece la relación entre el tiempo de ciclo de instrucción ( $T_{CY}$ ) y el tiempo de ciclo del ADC ( $T_{AD}$ ).

El convertidor necesita un ciclo  $T_{AD}$  para calcular cada uno de los bits del resultado, y un par de ciclos más para estabilizar el resultado. Eso implica que para muestras de 10 bits necesita 12 ciclos (para muestras de 12 bits, 14 ciclos). Al finalizar el tiempo de digitalización, el resultado se carga en el registro ADCxBUF0 o en DMA y la CPU generará una interrupción si está programada para ello. Finalizada la conversión el S&H puede volver a conectarse a un pin de entrada para dar comienzo a una nueva conversión.

En lo referente a los canales, el canal 0 (CH0) es el más flexible de cara a la selección de las entradas, ya que las 32 entradas digitales están conectadas a él (en el caso del ADC1; en el ADC2, sólo 16). El campo CH0SA (5 bits en el registro ADxCHS0) selecciona la entrada analógica que se conecta al canal.

## **Petición de interrupción del módulo ADC**

Es posible generar una petición de interrupción al finalizar cada conversión, o después de múltiples conversiones (hasta 16), según el valor que se les dé a los bits SMPI<3:0> (registro ADxCON2<5:2>). Si no se usa DMA el campo SMPI debe ser 0 para que se genere una interrupción al final de cada conversión.

## Información complementaria. Otras opciones de funcionamiento del ADC.

### Muestreo simultáneo.

Puede haber aplicaciones en las que resulte necesario muestrear varias señales en el mismo instante de tiempo. El bit de control SIMSAM (ADxCON1<3>) trabaja conjuntamente con diversos bits para llevar el control de la secuencia de muestreo y conversión de los canales.

Opciones:

CHPS<1:0>	SIMSAM	Secuencia de muestreo y retención	Ciclos de muestreo y conversiones por completar
00	X	Muestrea CH0, conversión CH0	1
01	0	Muestreo CH0, conversión CH0 Muestreo CH1, conversión CH1	2
1x	0	Muestreo CH0, conversión CH0 Muestreo CH1, conversión CH1 Muestreo CH2, conversión CH2 Muestreo CH3, conversión CH3	4
01	1	Muestreo simultáneo CH0 y CH1 Conversión CH0 Conversión CH1	1
1x	1	Muestreo simultáneo CH0, CH1, CH2 y CH3 Conversión CH0 Conversión CH 1 Conversión CH2 Conversión CH3	1

El bit de control SIMSAM no tiene ningún efecto en el funcionamiento del módulo si el par de bits CHPS<1:0> = 00.

En el CAD se puede iniciar el muestreo de forma manual o de forma automática.

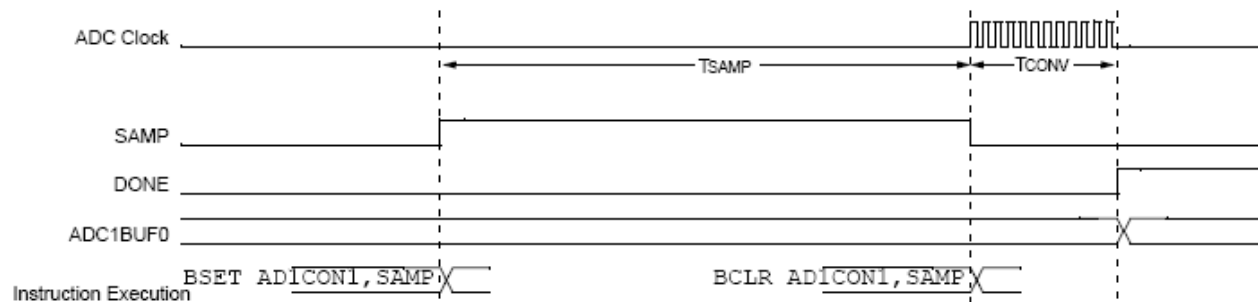
Activando el bit de control SAMP situado en ADCON1<1>, comienza el muestreo del CAD. El muestreo no se reanuda hasta que no se vuelva a poner a '1' el bit SAMP.

### Comienzo de la conversión de forma automática.

Activando el bit ASAM (registro ADCON1) comienza el muestreo en un determinado canal de forma automática, siempre que en ese canal no se esté realizando una conversión. El bit SIMSAM especifica si el muestreo es secuencial o simultáneo: si es secuencial el muestreo de un canal sólo se reanuda cuando finaliza la conversión del dato anterior; y si es simultáneo, el nuevo muestreo sólo se realiza cuando ha acabado la conversión de todas las muestras simultáneas.

Los bits de control SSRC (registro ADCON1) son los encargados de seleccionar la fuente de disparo de la digitalización, poniendo fin al muestreo. Dicho evento dará comienzo a la secuencia de conversiones previamente seleccionada. La modificación del campo SSRC debe realizarse cuando el módulo no esté operando, de modo que si el usuario quisiera cambiar su valor estando el módulo en funcionamiento, debería deshabilitarlo primero (bit ADON=0).

Cuando SSRC=0 el disparo de la conversión queda bajo control software, es decir, poniendo el bit de control SAMP a 1 comenzará el muestreo y poniéndolo posteriormente a 0 se finalizará el muestreo y comenzará la conversión. El Ejemplo 2 muestra esta situación.



## EJEMPLO 2

```
AD1PCFGL = 0xFFFFB;      // Los pines puerto B digitales, menos RB2 (análog.)
AD1CON1 = 0x0000;        // SAMP=0 finaliza muestreo y comienza conversión.
AD1CHS0 = 0x0002;        // Conecta RB2/AN2 como entrada del CH0.
AD1CSSL = 0;
AD1CON3 = 0x0002;        // Muestreo manual TAD = 2 Tcy (interno).
AD1CON2 = 0;

AD1CON1bits.ADON = 1;    // Encendido del módulo.

while (1)                // Bucle infinito.
{
    ADCON1bits.SAMP = 1;  // Empieza muestreo.
    DelayNmSec(100);      // Retardo de 100 ms.
    ADCON1bits.SAMP = 0;  // Empieza la conversión.

    while (!ADCON1bits.DONE); // ¿Se ha realizado la conversión?

    ADCValue = ADC1BUF0;   // Sí. Dejamos en ADCValue el resultado.
}
```

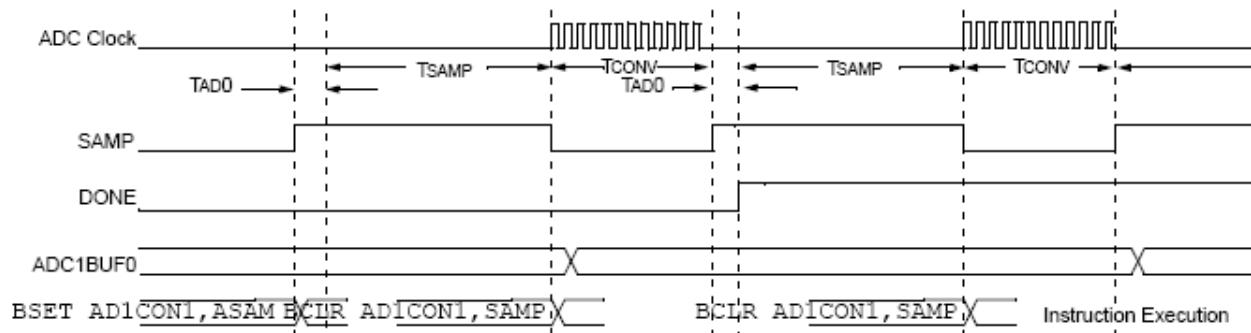
## Alternancia de canales.

Mediante el bit de ALTS (registro ADxCON2) se puede especificar que el módulo ADC alterne las sucesivas muestras o capturas entre dos series de entradas seleccionadas. Cuando el bit ALTS toma valor '1', el módulo alterna entre las entradas del Sample A (MUXA) en un muestreo y las entradas de MUXB en el siguiente. Las entradas especificadas por CH0SA<4:0>, CH0NA, CH123SA y CH123NA<1:0> se seleccionan como Sample A (MUXA), mientras que las entradas especificadas por CH0SB<4:0>, CH0NB, CH123SB y CH123NB<1:0> se seleccionan como Sample B (MUXB).

En los canales 1, 2 y 3 se puede elegir entre dos grupos de 3 entradas. El bit CH123SA (registro ADxCHS123) selecciona la fuente de las entradas analógicas positivas de los canales 1, 2 y 3. Por otro lado, los bits CH123NA (registro ADxCHS123) seleccionan la fuente de las entradas analógicas negativas de los canales 1, 2 y 3.

Cuando se utiliza sólo un canal S/H o muestreo simultáneo, SAMC debe programarse por lo menos para un pulso. Cuando utilizamos múltiples canales S/H con muestreo secuencial, podemos poner SAMC a '0', cero ciclos de reloj, obteniendo la menor tasa de conversión posible.

En el ejemplo 3 se activa el bit ASAM, de manera que se habilita el muestreo automático, de forma que poniendo el bit SAMP a '0' se fuerza el final del muestreo y el comienzo de la conversión. Además, el bit SAMP se pone a '1' automáticamente, al comienzo del intervalo de muestreo, siendo el software el encargado de controlar la puesta a '0' de este bit.



### EJEMPLO 3

```
AD1PCFG = 0xFF7F;           // Los pines del puerto B digitales, menos RB7
AD1CON1 = 0x0004;           // ASAM=1, el muestreo comenzará inmediatamente
                             // después de que finalice la conversión.
AD1CHS0 = 0x0007;           // Conecta RB7/AN7 como entrada del CH0.
AD1CSSL = 0;
AD1CON3 = 0x0002;           // Tiempo de muestreo manual, TAD = 2 Tcy.
AD1CON2 = 0;

AD1CON1bits.ADON = 1;       // Encendido del módulo

while (1)                   // Bucle infinito
{
    DelayNmSec(100);         // Retardo de 100 ms.
    ADCON1bits.SAMP = 0;     // Empieza la conversión
    while (IADCON1 bits.DONE); // Espera al fin de la conversión
    ADCValue = ADCBUFO;      // Se deja en ADCValue el resultado
}
```