

Microcontrolador dsPIC

Introducción

Este documento es una introducción a los microcontroladores dsPIC de Microchip. Antes de comenzar debemos definir que es un microcontrolador y en que se diferencia de un microprocesador. Un microcontrolador es un microprocesador con un alto nivel de especialización, optimizado para el control de dispositivos o equipos electrónicos. Normalmente son más pequeños pero integran muchos periféricos dentro del mismo circuito integrado o chip.

Las diferencias que generalmente se encuentran entre microprocesadores y microcontroladores son las que se muestran en la siguiente tabla:

Microprocesador	Microcontrolador
Alto nivel de generalización	Alto nivel de especialización
No integra periféricos dentro del mismo chip	Integra periféricos dentro del chip
Alto Costo	Bajo Costo
Velocidades de reloj altas	Velocidades de reloj medianas o bajas
Generalmente son CISC	Generalmente son RISC
Se utiliza para procesar información	Se utiliza para controlar dispositivos electrónicos
Necesita de gran cantidad de componentes externos para funcionar	Necesita de pocos componentes externos para funcionar

Los microcontroladores se utilizan en un sin número de aplicaciones, generalmente, asociadas al control de dispositivos electrónicos. Como ejemplos de utilización se pueden mencionar los siguientes:

- Control de motores: continuos, PWM, etc.
- Biometría: Reconocimiento huellas digitales y voz, etc.
- Telefonía: detección de DTMF, identificadores de llamadas
- Aparatos electrónicos: impresoras, scanners, módems, microondas, etc.
- Mediciones en general: temperatura, humedad, señales biológicas.
- Procesamiento de señales: reducción de ruidos.
- Automotriz: airbag,
- Juguetes y Robótica.

Arquitectura

La arquitectura tradicional de computadoras y microprocesadores se basa en el esquema propuesto por John Von Neumann, en el cual la unidad central de proceso, o CPU, esta conectada a una memoria única que contiene las instrucciones del programa y los datos (ver figura 1).

La arquitectura conocida como Harvard, consiste simplemente en un esquema en el que el CPU esta conectado a dos memorias por intermedio de dos buses separados (ver figura 2). Una de las memorias contiene solamente las instrucciones del programa, y es llamada Memoria

de Programa. La otra memoria solo almacena los datos y es llamada Memoria de Datos. Ambos buses son totalmente independientes y pueden ser de distintos anchos. Para un procesador de Set de Instrucciones Reducido, o RISC (Reduced Instrucción Set Computer), el set de instrucciones y el bus de la memoria de programa pueden diseñarse de manera tal que todas las instrucciones ocupen una sola posición de memoria de programa de longitud. Además, como los buses son independientes, el CPU puede estar accediendo a los datos para completar la ejecución de una instrucción, y al mismo tiempo estar leyendo la próxima instrucción a ejecutar. Por lo tanto se puede determinar que las principales ventajas de este tipo de arquitectura son dos:

- el tamaño de las instrucciones no esta relacionado con el de los datos, y por lo tanto puede ser optimizado para que cualquier instrucción ocupe una sola posición de memoria de programa, logrando así mayor velocidad, menor longitud de programa y simplicidad.
- el tiempo de acceso a las instrucciones puede superponerse con el de los datos, logrando una mayor velocidad de operación.

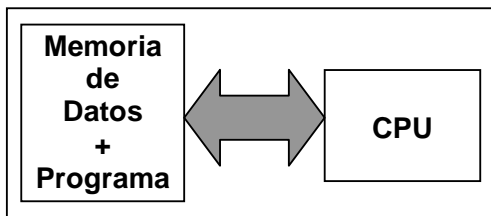


Figura 1. Arquitectura Von Neumann

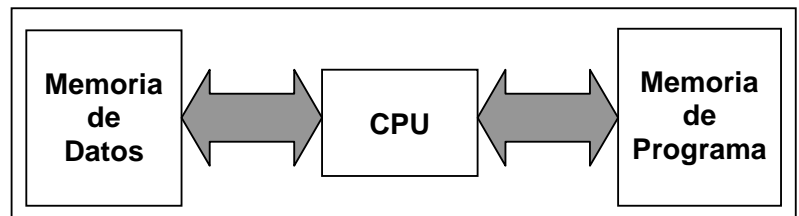


Figura 2. Arquitectura Harvard

Una pequeña desventaja de los microcontroladores o microprocesadores con arquitectura Harvard es el acceso a constantes y tablas de valores. Las constantes son similares a los datos, por lo que deberían estar en la memoria de datos, algo que no es posible ya este tipo de memoria es volátil (RAM) y se perderían sus valores. La solución es ubicar las constantes en memoria de programa, que es no volátil (normalmente ROM, OTP, EPROM o FLASH), proveyendo un conjunto de instrucciones adicionales para poder leer parte de esta memoria como si fuera de datos. A este tipo de arquitectura se la denomina Harvard Modificada.

La arquitectura de los microcontroladores dsPIC esta basada en una arquitectura Harvard Modificada con memoria de datos de 16 bits y memoria de programa de 24 bits. Tiene un juego de instrucciones reducido (RISC) donde la mayoría de las instrucciones se ejecutan en un ciclo de instrucción. Una característica que diferencia a los dsPIC de otros microcontroladores es que tiene un soporte para el procesamiento de señales digitales (DSP). También tiene un integrado multiplicador y soporte para divisiones.

Como todos los microcontroladores, los dsPIC, necesitan pocos componentes externos para su funcionamiento y tienen integrado en el chip un gran número de periféricos de características muy diversas. Estos dispositivos periféricos integrados se comunican con el exterior a través de los pines del chip y, como normalmente los comparten, se deben configurar para utilizar solo un dispositivo periférico dejando inactivos a los demás. Mas aún, también es posible desactivar un conjunto de periféricos y utilizar los pines como entradas/salidas digitales.

Los dsPIC pueden integrar los siguientes módulos de dispositivos periféricos:

- Temporizadores.
- Comparadores de salida.
- Capturadores de entrada
- Comunicación:
 - SPI (Serial Peripheral Interface).
 - I²C (Integrated Intercommunication Circuit).
 - UART(Universal Asynchronus Receiver Transmitter).
 - CAN (Control Area Network).
 - DCI.
- Control de Motores:
 - PWM (Pulse Wave Modulation).
 - Codificadores de Cuadratura.
- Conversores Analógicos digitales.
- DMA (Acceso Directo a Memoria).

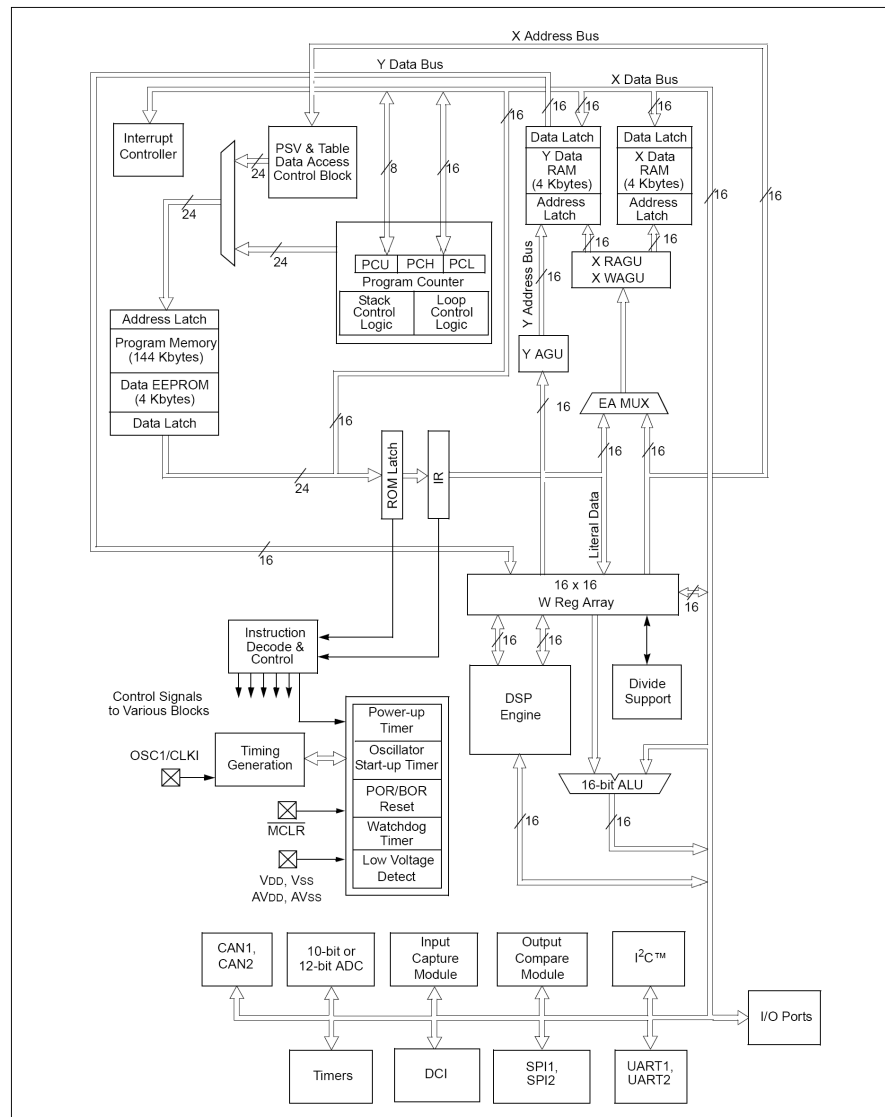


Figura 3. Diagrama de los bloques de hardware del microcontrolador.

Actualmente en el mercado existen dos líneas de dsPIC. La línea dsPIC30F, mas general, y la línea dsPIC33F, para aplicaciones de mayor rendimiento, con un conjunto de dispositivos integrados más innovadores. Dentro de cada línea existen varios modelos que se diferencian por:

- El conjunto dispositivos periféricos diferentes que integran.
- La cantidad de dispositivos de cada tipo de dispositivo periférico.
- La cantidad y el tipo de memoria de programa.
- La cantidad y tipo de memoria de datos.
- Cantidad de pines.
- Empaquetamiento del chip.
- Velocidades de trabajo.

Memoria de Programa

Los dispositivos dsPIC tienen un espacio de memoria de programa direccionable de 4M de instrucciones, con 24bits de instrucción. Provee tres métodos para acceder al espacio de memoria de programa:

1. A través de los 23 bits del contador de Programa.
2. A través de dos instrucciones para leer (TBLRD) y escribir tablas (TBLWT).
3. a través del mapeo de segmentos de programa de 32 kbytes dentro del espacio de memoria.

La memoria de programa se divide en el espacio de programa de usuario y el espacio de configuración del usuario. El espacio de programa de usuario contiene un vector de Reinicialización (Reset vector), tablas de vectores de interrupción, memoria de programa y memoria de datos EEPROM. El espacio de configuración de usuario contiene bits de configuración, no volátiles ,para la configuración de opciones y localización de dispositivos.

Se provee un conjunto de instrucciones para permitir mover un byte o un word entre el espacio de memoria de programa y el espacio de memoria de datos. Las instrucciones de lecturas de tablas son las que permiten transferir datos de la memoria de programa a la de datos. Las instrucciones de escritura de tablas son las que permiten transferir datos de la memoria de datos a un sector especial de memoria de programa, de tipo EEPROM o flash.

Hay cuatro instrucciones que permiten acceder a tablas situadas en memoria de programa. Estas instrucciones acceden de a 16 bits pero hay variantes que permiten quedarse solo con 8 bits. Estas instrucciones son:

Instrucción de Tabla	Descripción
TBLRDL	leer los 16 bits bajos de la zona de memoria
TBLWTL	escribir los 16 bits bajos de la zona de memoria
TBLRDH	leer los 16 bits altos de la zona de memoria (8 bits altos en cero)
TBLWTH	escribir los 16 bits altos de la zona de memoria (8 bits altos en cero)

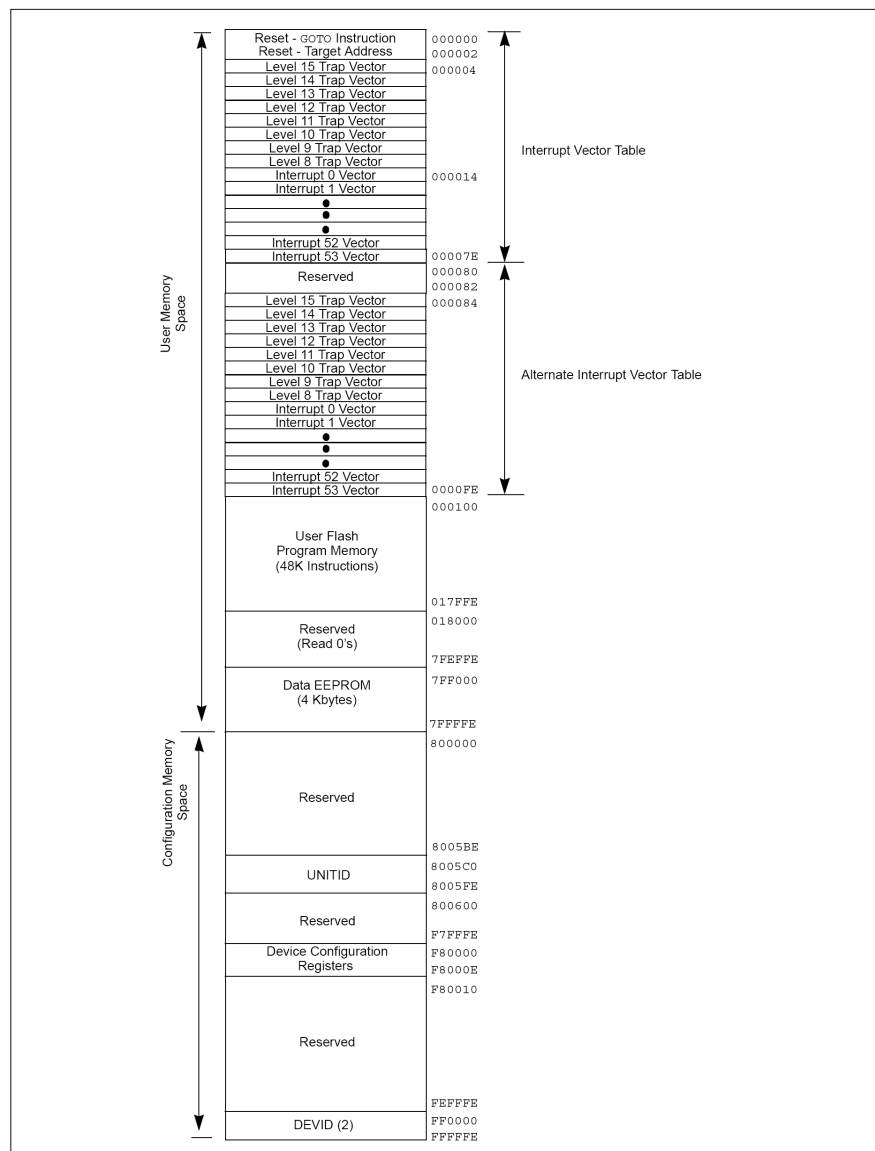


Figura 4. Esquema de memoria de programa.

Interrupciones

El microcontrolador dsPIC posee un modulo de control de interrupciones que reduce los numerosos pedidos de interrupción de los periféricos a un simple pedido de interrupción para que procese la CPU. A continuación se mencionan algunas de sus características que varían según la línea del microcontrolador:

- Hasta 8 excepciones de procesador y trampas (traps) de software.
- 7 niveles de prioridad seleccionadas por el usuario.
- Tabla de Vector de Interrupciones (IVT) de hasta 62 entradas.
- Un único vector para cada fuente de interrupción o excepción.
- Tabla de Vector Alternativa (AIVT) para soporte de depuración.
- Latencia fija para entradas y regresos de interrupción.

La tabla del vector de interrupción (IVT) se encuentra en la memoria de programa a partir de la posición 0x000004. Contiene 62 vectores que consisten de 8 vectores para trampas (traps), no-enmascarables, mas 54 vectores para fuentes de interrupción.

La tabla del vector de interrupción alternativa (AIVT) se encuentra localizada luego del IVT. Mediante un bit del registro de control de interrupciones denominado INTCON2 se indica al microcontrolador que se utilizará el vector de interrupciones alternativo para depuración o para emulación.

Memoria de Datos

El ancho de los datos de la memoria, al igual que los registros de trabajos son de 16 bits. El espacio de memoria esta separado en dos partes. Estos bloques pueden ser accedidos separadamente de forma simultanea, o como si fueran un único bloque. Esto permite que varias instrucciones relacionadas con el uso del DSP puedan obtener un mejor rendimiento.

Las direcciones de memoria comprendidas entre 0x000 y 0x7FFF están reservadas como registros para funciones especiales, relacionadas bits de control y estado de dispositivos asociados a la CPU y demás periféricos.

La memoria ram comienza a partir de la dirección 0x800 y se divide en dos bloques, denominados espacio de datos X e Y. En las instrucciones típicas que realiza la MCU siempre se accede al espacio formado por X e Y como si fuera un único espacio. Mientras que en las instrucciones típicas que realiza el DSP accede separadamente al espacio de datos de X y de Y para recuperar los operandos de forma simultanea, lo que permite un mejor rendimiento.

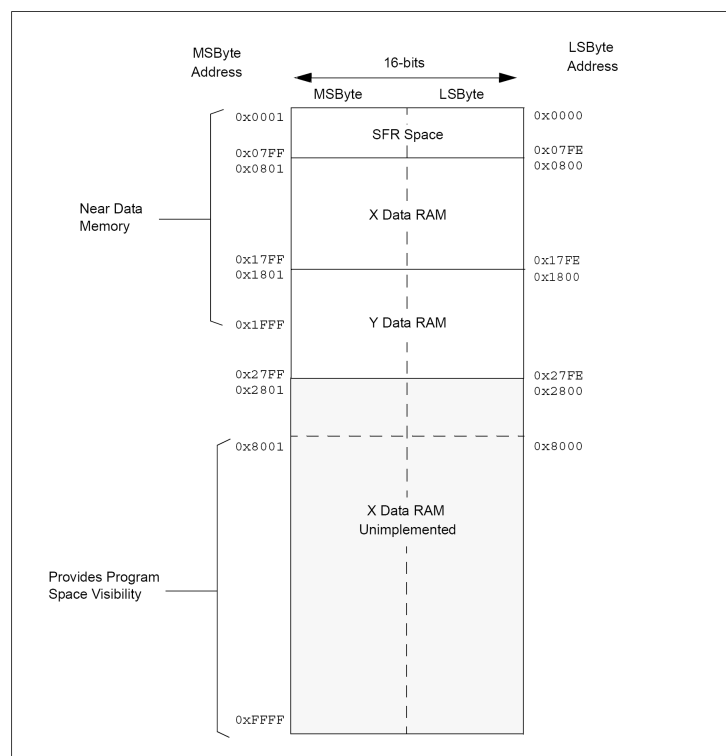


Figura 5. Esquema de memoria de datos.

Direccionamientos

El dsPIC provee una gran flexibilidad para el direccionamiento de la memoria de datos. Básicamente se pueden dividir en direccionamiento básico, para instrucciones comunes, y direccionamiento de DSP, para instrucciones específicas de DSP.

Direccionamiento Básico

Existen cuatro tipos de direccionamientos básicos:

- Registro de Archivo: este direccionamiento es usado por instrucciones que usan la dirección de un registro predeterminado. Es decir que estas instrucciones tienen un operando implícito, que es un registro de trabajo en particular (W0).
- Directo por Registro: este direccionamiento es utilizado en las instrucciones para obtener un operando que reside en un registro de trabajo.
- Indirecto por Registro: este direccionamiento es utilizado en las instrucciones para obtener un operando de la dirección de memoria que reside en un registro de trabajo. Este modo de direccionamiento provee variantes que permiten obtener diferentes direcciones de operandos a partir del contenido de un registro. Estas variantes son:
 - Pre-incremento: incrementa el registro en 2 antes de obtener la dirección.
 - Pre-decremento: decrementa el registro en 2 antes de obtener la dirección.
 - Post-incremento: incrementa el registro en 2 luego de obtener la dirección.
 - Post-decremento: decrementa el registro en 2 luego de obtener la dirección.
 - Desplazamiento por registro: obtiene la dirección del operando sumando 2 registros, sin alterarlos.
 - Sin modificación: usa directamente el contenido del registro para obtener la dirección del operando.
 - Literal: obtiene la dirección del operando sumando una constante al registro en cuestión. La constante puede ser como máximo un entero de 10 bits.
- Inmediato: el operando es una constante incluida en la instrucción.

Direccionamiento de DSP

Existen tres tipos de direccionamientos para instrucciones que usan el DSP:

- Registro de Archivo: este direccionamiento es usado por instrucciones que usan la dirección de un registro predeterminado. Es decir que estas instrucciones tienen un operando implícito, que es un registro de trabajo en particular (W0).
- Indirecto: es utilizado para obtener un operando de la dirección de memoria que reside en un registro de trabajo. Este modo de direccionamiento provee variantes que permiten obtener diferentes direcciones de operandos a partir del contenido de un registro. Estas variantes son:
 - Post-incremento: incrementa el registro en 2, 4 o 6 luego de obtener la dirección.
 - Post-decremento: decrementa el registro en 2, 4 o 6 luego de obtener la dirección.
 - Desplazamiento por registro: obtiene la dirección del operando sumando 2 registros, sin alterarlos.
 - Sin modificación: usa directamente el contenido del registro para obtener la dirección del operando.

Direccionamientos especiales

El dsPIC provee soporte especial para los tipo de direccionamiento indirecto, que generalmente se utiliza en algoritmos de DSP. En las siguientes variantes para el direccionamiento indirecto:

- Direccionamiento de módulo: el tipo de direccionamiento de módulo (o circular) provee soporte automático por hardware para implementar buffers de datos circulares, evitando la necesidad de validar los límites por software y mejorando el rendimiento de los algoritmos.
- Direccionamiento de bits-revertidos: este tipo de direccionamiento permite acceder a elementos de buffers de manera no lineal. El mismo consiste en revertir una secuencia de bits de derecha a izquierda, facilitando el acceso a elementos en ciertos algoritmos DSP, como la FFT (Fast Fourier Transform), y reduciendo el tiempo de procesamiento significativamente.

Registros

Registros de Trabajo

Los 16 registros de trabajo (W_x , donde x es un número entre 0 y 15) pueden funcionar como registros de datos, registro de direcciones y registros de desplazamiento de direcciones. La función de un registro de trabajo W está determinada por el modo de direccionamiento de la instrucción en la cual es utilizado. A continuación se muestran ejemplos de instrucciones donde intervienen registros de trabajo como datos y direcciones:

```
MOV    W0,  W1           ; mueve el contenido de W0 a W1
MOV    W0, [W1]          ; mueve W0 a la dirección contenida en W1
ADD    W0, [W4], W5       ; suma el contenido de W0 al contenido
                           ; apuntado por W4. Guarda el resultado en W5
```

Todos los registros tienen 16 bits. Para aquellas instrucciones que intervienen operandos de 8 bits solo se toman en cuenta 8 bits menos significativos del registro afectado por la instrucción. Todos los registros están mapeados en memoria por lo que es posible leerlos o escribirlos accediendo a la memoria de datos. Un ejemplo de este caso puede ser:

```
MOV    0x0004, W10        ; equivale a MOV W2, W10
```

Registro de Pila

El registro de trabajo $W15$ o SP (Stack Pointer) sirve como puntero de pila dedicado y es modificado automáticamente por procesamiento de excepciones, llamados y retorno de subrutinas. Es posible manipularlo ya que como todos los registros de trabajo se encuentra mapeado en memoria. Para evitar posibles desalineaciones en el acceso a memoria, que debe ser de a 16bits, el bit menos significativo es puesto a cero por hardware.

Inicialmente el registro de pila apunta a la primera posición de memoria libre. Cuando se ejecuta una instrucción *PUSH origen* se guardan 16 bits de *origen* en la posición de memoria apuntada por SP y luego se lo decrementa, apuntando nuevamente a una posición de memoria

libre. Cuando se ejecuta la instrucción *POP destino*, SP se decrementa apuntando a la posición de memoria donde se almacenaron los últimos 16 bits para luego copiarlos en *destino*.

Registro Limitador de Pila

Existe un registro adicional de 16 bits asociado al registro de pila, denominado *SPLIM*, que limita el valor máximo que puede alcanzar el puntero de pila. Si el puntero de pila es superior al valor de *SPLIM* en dos posiciones o es inferior al valor 0x800 se produce un error de pila (stack error) generando una excepción.

Registros Acumuladores

El dsPIC cuenta con dos registros acumuladores de 40 bits utilizados por instrucciones DSP que permiten realizar operaciones matemáticas y desplazamientos lógicos. Cada acumulador esta mapeado en los siguientes 3 registros físicos:

- ACCxU (bits 39–32)
 - ACCxH (bits 31–16)
 - ACCxL (bits 15–0)
- (x identifica el acumulador A o B correspondiente)

Registro Contador de Programa

El contador del programa (PC) tiene un ancho de 23 bits. A través de este registro es posible direccionar 4M de instrucciones de 24 bits en el espacio de memoria de programa. El bit menos significativo esta puesto a 0 para proveer compatibilidad con el espacio de direccionamiento de datos que es 16 bits de ancho(2 bytes). Esto implica que durante la ejecución normal de las instrucciones el contenido del PC se incremente de 2 en 2.

Registro TBLPAG

El registro TBLPAG es un registro destinado a almacenar los 8 bits mas significativos de la dirección de memoria para las instrucciones de lectura y escritura de tablas. Este tipo de instrucciones se utilizan para transferir información entre el espacio de memoria de programa y el de datos.

Registro PSVPAG

El registro PSVPAG (Program Space Visibility Page) permite mapear una sección del espacio de memoria de programa de 32 Kbytes en los 32 kbytes mas altos del espacio de memoria de datos. Esta característica ofrece acceso transparente para datos constantes.

Registro RCOUNT

El registro RCOUNT es un registro de 14 bits que mantiene la cuenta de bucles para la instrucción de repetición REPEAT (repite solo una instrucción). Cuando se ejecuta esta instrucción con un parámetro, este se carga en el registro RCOUNT y se comienza la ejecución de un bucle RCOUNT +1 veces.

Registro DCOUNT

El registro DCOUNT es un registro de 14 bits que mantiene la cuenta de bucles para la instrucción de repetición DO (repite un bloque de código). Cuando se ejecuta esta instrucción con un parámetro, este se carga en el registro DCOUNT y se comienza la ejecución de un bucle DCOUNT +1 veces.

Registro DOSTART

El registro DOSTART contiene la dirección de inicio de un bucle de repetición DO por hardware. Cuando se ejecuta una instrucción DO, el registro DOSTART es cargado con la dirección siguiente a la instrucción DO. Cuando se activa el bucle el programa comienza la ejecución por la dirección que indica el registro DOSTART.

Registro DOEND

El registro DOEND contiene la dirección de fin de un bucle de repetición DO por hardware. Cuando se ejecuta una instrucción DO, el registro DOEND es cargado con la dirección especificada en la instrucción DO. Cuando se activa el bucle y la ejecución del programa llega a la dirección que indica el registro DOEND se continua con la instrucción indicada por el registro DOSTART.

Registro de Estado

El registro de estado es un registro de 16 bits que mantiene información acerca del estado de las instrucciones que se han ejecutado recientemente. Los bits de este registro pueden agruparse según su función de la siguiente manera:

- Bits de Estado MCU de la Unidad Aritmético-Lógica: estos bits son afectados por las operaciones aritméticas y lógicas del microcontrolador. Los bits que componen este grupo son de Acarreo, Cero, Desbordamiento, Negativo y Acarreo de Dígito denominados C, Z, OV, N y DC respectivamente.
- Bits de Estado de repetición: este grupo lo componen 2 bits que indican si los bucles de repetición están activos. Los bits se denominan DA para el bucle de repetición DO y RA para el bucle de repetición REPEAT.
- Bits de Estado DSP de la Unidad Aritmético-Lógica: estos bits son afectados por las instrucciones DSP. Los bits que componen este grupo están asociados a los registros acumuladores A y B e indican si la operación DSP produjo saturación, o desbordamiento en el acumulador A, en el B o en los dos. Estos 6 bits son denominados SA, SB, OA, OB, SAB y OAB.
- Bits de estado del nivel de prioridad de interrupción: este grupo esta compuesto por 3 bits que indican el nivel de la prioridad que tiene que tener una interrupción para poder interrumpir al procesador. Con estos 3 bits se puede indicar 8 niveles de interrupción, de 0 a 7, donde 0 es el nivel más bajo.

Registro de Control de Núcleo

Este registro de control, denominado CORCON, que es usado para controlar la configuración del dsPIC. Este registro provee las siguientes funcionalidades:

- mapeo de espacio de memoria de programa en memoria de datos (bit PSV).
- Seleccionar el modo de saturación de acumuladores A y B (bits SATA y SATB).
- Seleccionar saturación de escritura en espacio de datos (bits SATDW).
- Seleccionar modos de Saturación y Redondeo de acumuladores (bits ACCSAT y RND).
- Seleccionar modo de multiplicación para operaciones DSP que puede ser fraccionaria, entera, con signo, sin signo (bits US e IF).
- Abortar ciclo de repetición DO prematuramente (bits DL).

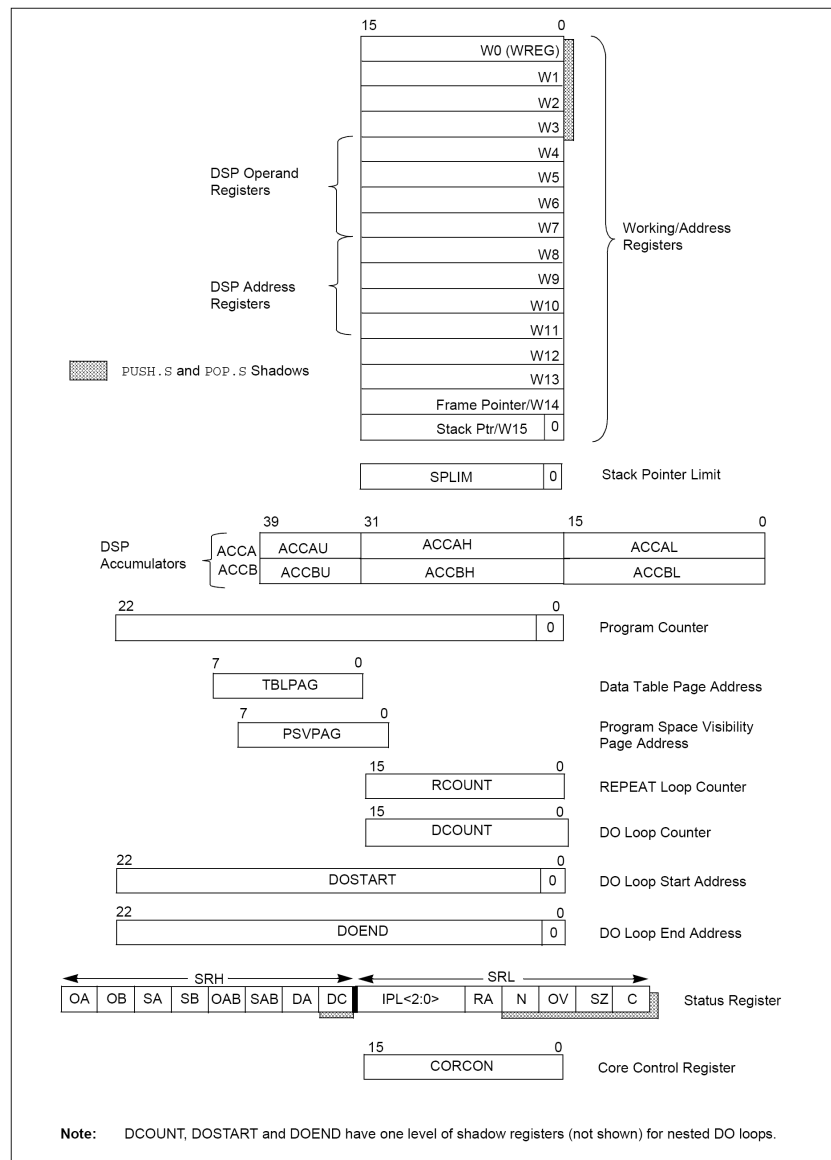


Figura 6. Principales registros del microcontrolador.

Procesador de Señales Digitales (DSP)

La función principal de un DSP es dar soporte para el procesamiento de señales de naturaleza digital. El DSP del microcontrolador es un bloque de hardware que se encuentra alimentado a través de los registros de trabajo (W) y que cuenta registros especializados propios.

La mayoría de las instrucciones del DSP están diseñadas para un rendimiento óptimo en tiempo real. Por ejemplo, este conjunto de instrucciones específicas del DSP permiten la búsqueda concurrente de dos operandos en memoria mientras se están multiplicando dos registros de trabajo y realizando la acumulación del resultado, todo en un ciclo de instrucción. Los componentes que caracterizan al DSP son de alta velocidad y comprenden:

- Multiplicador de 17 bits por 17 bits de alta velocidad.
- Desplazador (barrel shifter).
- Sumador / restador de 40 bits.
- 2 acumuladores de 40 bits.
- Lógica de redondeo y saturación.

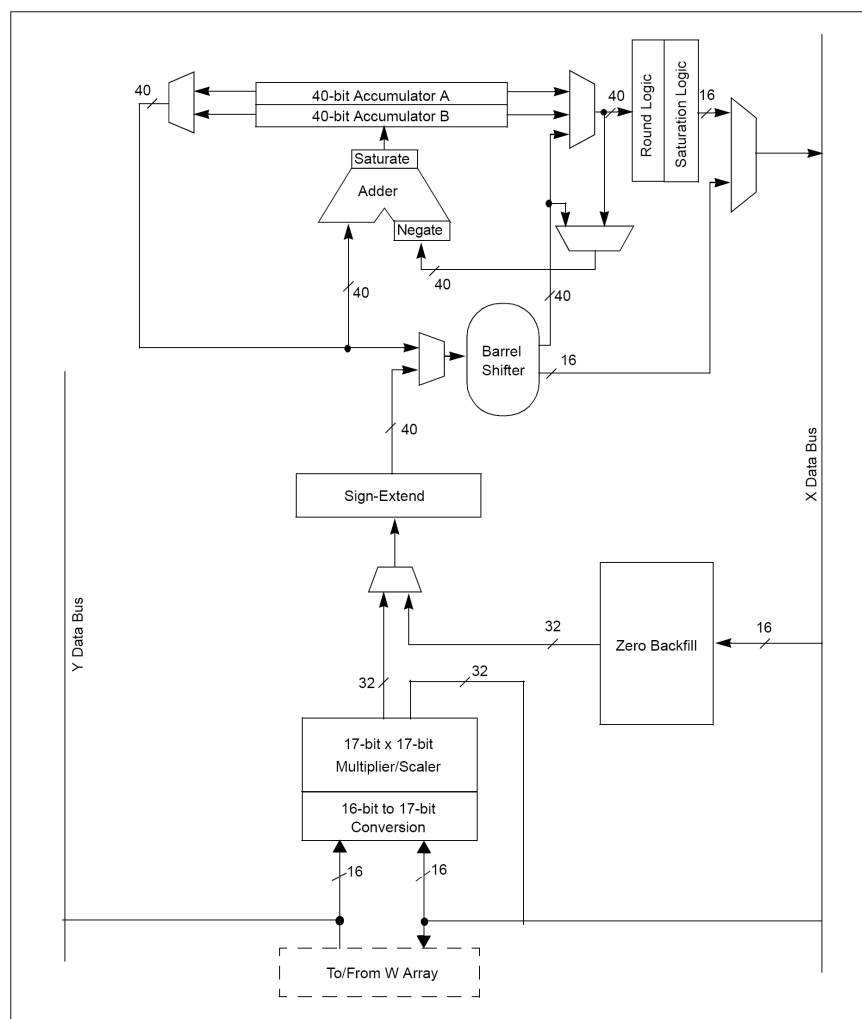


Figura 7. Diagrama del bloque de hardware del DSP.

Multiplicador

El dsPIC posee un multiplicador de 17 bits x 17 bits permite multiplicar operandos enteros o fraccionarios de 16 bits con signo, sin signo y mixtos, combinándolos. El resultado es expresado como entero de 32 bits o fraccionario Q.31 (1.31).

El multiplicador toma 16 bits de entrada y los convierte en 17 bits. En el bit más significativo se extiende el bit de signo si el operando es negativo, mientras que si el operando no tiene signo se coloca un cero en este bit. El bit 0 del registro de control CORCON determina si el tipo de multiplicación a realizar por el multiplicador es entera o fraccional.

El multiplicador es utilizado tanto por el DSP como por la MCU de la ALU. Ejemplos de las instrucciones:

Instrucción DSP	Descripción	Equivalencia Algebraica
MAC	Multiplicar y sumar al acumulador Calcular cuadrado y sumar al acumulador	$A = A + B * C$ $A = A + B^2$
MSC	Multiplicar y Restar desde acumulador	$A = A - B * C$
MPY	Multiplicar	$A = B * C$
MPY.N	Multiplicar y negar resultado	$A = -B * C$
ED	Calcular distancia Euclídea parcial	$A = (B - C)^2$
EDAC	Calcular distancia Euclídea parcial y sumar al acumulador	$A = A + (B - C)^2$

Desplazador

El desplazador puede realizar desplazamientos de hasta 16 bits de izquierda a derecha y viceversa en solo un ciclo de instrucción. La mayoría de las instrucciones están diseñadas para un rendimiento óptimo en tiempo real. Estas instrucciones permiten la búsqueda concurrente de dos operandos a memoria mientras se están multiplicando dos registros W, realizando la acumulación del resultado, todo en un ciclo de instrucción.

Acumuladores

Existen dos acumuladores de 40 bits, denominados ACCA y ACCB, que se utilizan para los resultados obtenidos por las instrucciones DSP. Para las operaciones fraccionales que usan acumuladores el punto decimal se coloca a la derecha del bit 31, por lo que el rango representable se encuentra entre -256 y $256 \cdot 2^{-31}$. Para operaciones enteras que usan acumuladores el punto decimal se ubica a la derecha del bit 0, por lo que el rango que es posible almacenar se encuentra entre $-549,755,813,888$ y $549,755,813,887$.

Sumador/Restador de Acumuladores

Los registros acumuladores cuentan con un sumador/restador de 40 bits con lógica de extensión de signo para los resultados de multiplicaciones. Es posible seleccionar uno de los dos acumuladores como origen con pre-acumulación o como destino con post-acumulación.

El sumador/restador de 40 bits puede opcionalmente negar uno de sus operandos de entrada para cambiar el signo de su resultado (sin cambiar el operando).

Soporte de división

Los tipos de divisiones que soporta el microcontrolador son:

- 16/16 bits fraccional con signo.
- 16/16 bits con signo.
- 16/16 bits sin signo.
- 32/16 bits con signo.
- 32/16 bits sin signo.

El cociente para todas las instrucciones de división debe estar ubicado en el registro W0 y el resto en registro W1. El divisor de 16 bits como el dividiendo de 16 bits pueden estar ubicado en cualquier registro de trabajo W. El dividendo si es de 32 bits deberá estar ubicado en dos registros de trabajo adyacentes.

Todas las instrucciones de división son operaciones iterativas y deben ser ejecutadas en un ciclo de 18 repeticiones. El usuario es el responsable de esta programación utilizando las instrucciones especiales de repetición. Una operación completa de división toma 19 ciclos de instrucción para ejecutarse.

El ciclo de división es interrumpible como en cualquier otro ciclo de repetición. Todos los datos son restaurados en sus respectivos registros luego de cada iteración en el ciclo, por lo que el usuario es responsable de salvar los registros W apropiados durante una interrupción. Durante toda la operación el hardware de división almacena los resultados intermedios en los registros de trabajo correspondientes, por lo que los valores de estos registros solo tendrán significado para el usuario al finalizar el ciclo de repetición número 18.

Comunicación

Módulo SPI

El módulo de Interfase Periférica Serie (SPI, Serial Peripheral Interface) es una interfase serie que permite la comunicación con otros microcontroladores o dispositivos periféricos como memorias seriales EEPROM, registros de desplazamiento, manejadores de display, conversores A/D, etc. El módulo SPI es compatible con las interfases SPI y SIOP de Motorola.

El microcontrolador puede contar con varios módulos SPI, dependiendo de la línea a la que pertenezca. Cada puerto serie SPI consiste de los siguientes registros de funciones especiales disponibles para el usuario:

- SPIxBUF: registro de 16 bits utilizado como buffer de transmisión y recepción. La dirección de este registro está compartida por 2 registros internos, SPIxTXB de transmisión y SPIxRXB de recepción.
- SPIxCON: registro de control de 16 bits para configurar el modo de operación.
- SPIxSTAT: registro de estado de 16 bits que indica varias condiciones de estado.
(Nota: x indica el número del dispositivo SPI)

Este módulo cuenta con un registro interno de 16 bits que cumple la función de desplazamiento de los datos de entrada y de salida, denominado SPIxSR.

El reloj que utiliza el módulo SPI como generador de la frecuencia de transmisión es el reloj de ciclo de instrucción. Para poder variar la frecuencia de comunicación, la señal de reloj se utiliza como señal de entrada de un prescaler. La señal de salida de éste se conecta a un segundo prescaler para finalmente la señal de salida de reloj (SCKx) para transmitir y recibir datos.

En la siguiente figura se muestra el diagrama de bloques del módulo SPI.

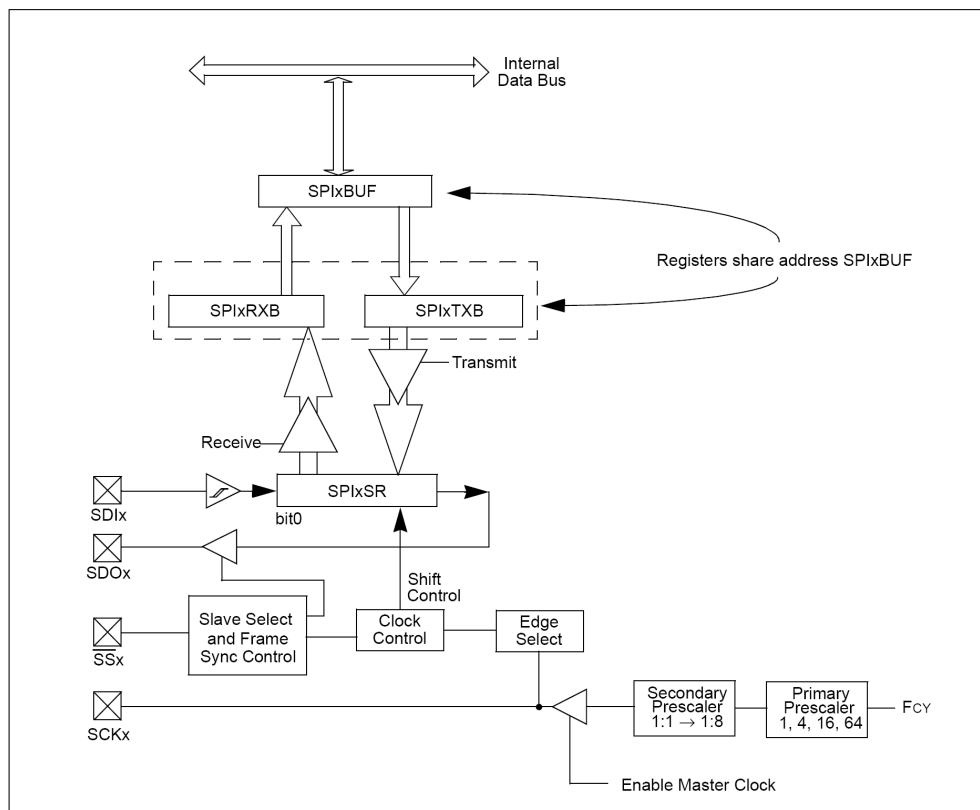


Figura 8. Diagrama de bloques del módulo SPI.

La interfase serie SPI tiene cuatro líneas de comunicación. Las funciones de cada línea son las siguientes:

- SDIx: entrada de datos serie (serial data input).
- SDOx: salida de datos serie (serial data output).
- SCKx: señal reloj para desplazamiento de entradas o salidas (serial clock).
- SSx: selección de esclavo (slave select) o pulso de E/S para sincronización de cuadro.

El módulo SPI puede ser configurado para trabajar con los siguientes modos de operación:

- Modo de Transmisión y Recepción de datos de 8 o 16 bits: el dispositivo puede configurarse para transmitir y recibir 8 o 16 bits. Es necesario 1 ciclo de reloj para cada bit.
- Modo de Trabajo Maestro o esclavo: cuando el dispositivo es programado como maestro genera la señal de reloj para transmitir o recibir datos al dispositivo esclavo.

Cuando es programado como esclavo toma la señal de reloj del maestro para transmitir o recibir datos.

- Modo de Trama (Frame) SPI: el dispositivo SPI puede configurarse para generar un pulso en la señal SS_k para sincronizar el inicio de la transmisión o de la recepción de cada dato.

Este módulo SPI, como los demás dispositivos periféricos de comunicación, permite configurar interrupciones para que se generen cada vez que se transmite o se recibe un dato.

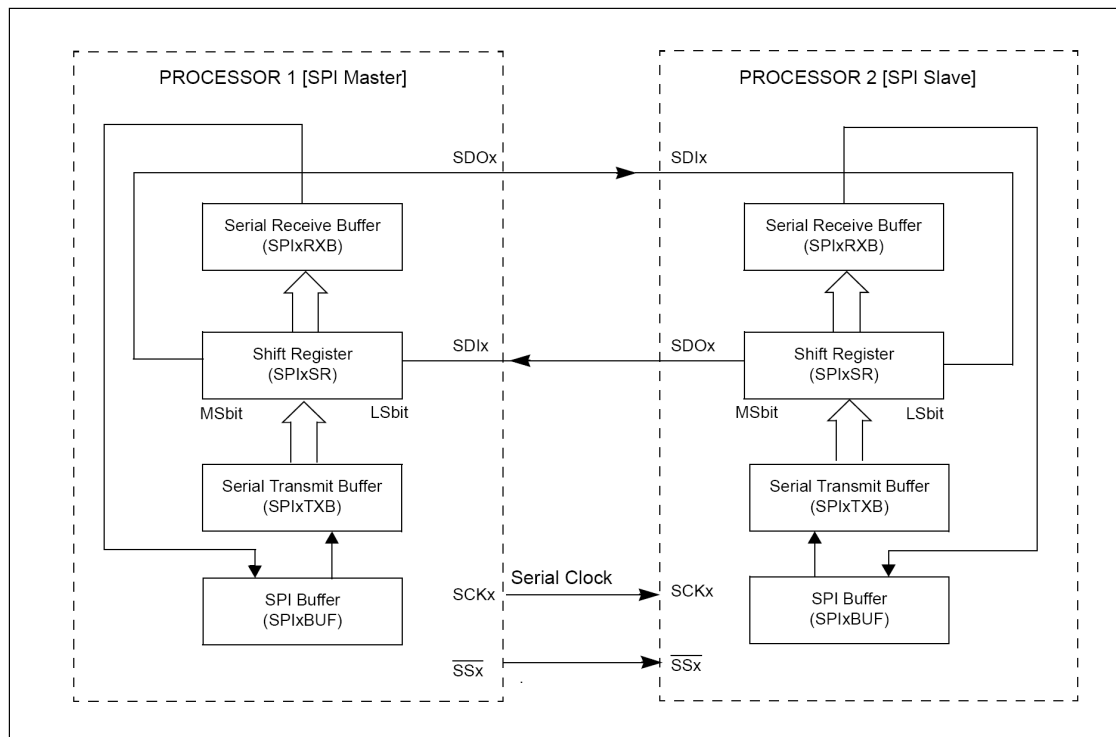


Figura 9. Ejemplo de conexión SPI Maestro/Esclavo.

Modulo I2C

El módulo I²C (Inter.-Integrated Circuit) es una interfase serie de dos líneas utilizada para la comunicación del microcontrolador con otros dispositivos periféricos u otros microcontroladores. Estos dispositivos periféricos pueden ser memorias EEPROM seriales, manejadores de display, conversores analógicos, etc.

La interfase I²C emplea un protocolo comprensivo para asegurar la confiabilidad tanto de la transmisión como de la recepción de datos. Cuando se realiza una transmisión, un dispositivo, el “maestro”, inicia la transferencia en el bus y genera las señales de reloj para sincronizar la transferencia. Mientras tanto los demás dispositivos que integran el sistema, los “esclavos”, reciben la transferencia para luego emitir una respuesta. La línea de comunicación utilizada como reloj se denomina “SCL” se utiliza como salida del dispositivo maestro y de entrada al dispositivo esclavo. La línea de comunicación utilizada para la transmisión y recepción de datos, denominada “SDA”, es bidireccional y la utilizan tanto el dispositivo maestro como los dispositivos esclavos.

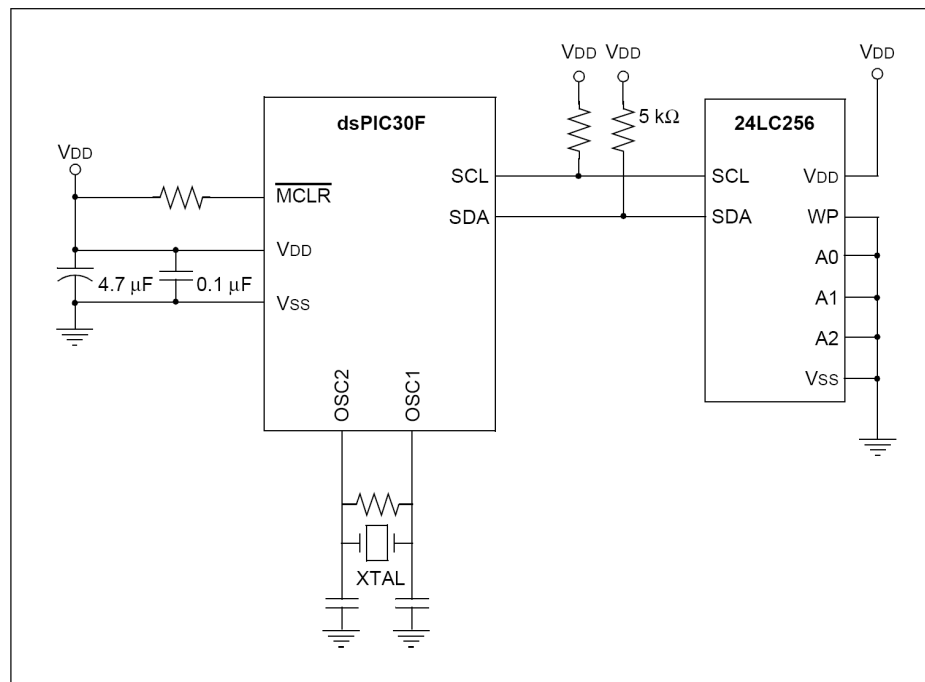


Figura 10. Conexión I²C con memoria EEPROM

En el protocolo de interfase I²C cada dispositivo que interviene en el sistema tiene una dirección asignada. Cuando el dispositivo maestro inicia una transferencia de datos, transmite primero la dirección del dispositivo esclavo que recibirá la transmisión. Cada vez que se inicia una transmisión todos los dispositivos del sistema (incluyendo el maestro) reciben la transferencia (“escuchan”) y verifican si la dirección enviada en la transmisión coincide con la suya. En el bit menos significativo de la dirección se le indica al dispositivo esclavo si la operación requerida por el dispositivo maestro es de lectura o escritura. En toda comunicación el maestro y el esclavo siempre trabajan de modos de operación opuestos durante la transferencia de datos. En otras palabras cuando el maestro transmite el esclavo recibe y viceversa. No obstante en cualquiera de los casos anteriores, el dispositivo maestro genera la señal de sincronización sobre la línea de reloj SCL.

El módulo I²C puede intervenir en cualquier sistema I²C de las siguientes maneras:

- como dispositivo esclavo.
- como dispositivo maestro.
- como dispositivo maestro/esclavo en un sistema multi-maestro, utilizando detección de colisiones y arbitraje.

El módulo I²C contiene lógica I²C independiente tanto para el maestro como para el esclavo, donde cada uno puede generar sus propias interrupciones basadas en sus eventos. En los sistemas multi-maestro el software se divide simplemente en un controlador maestro y un controlador esclavo.

Cuando la lógica de maestro esta activa, la lógica de esclavo permanece también activa detectando el estado del bus y recibiendo potenciales mensajes tanto del sistema como de sí mismo. En los sistemas multi-maestro, el módulo I²C permite la detección conflictos de bus por colisiones y provee métodos para terminar y reiniciar los mensajes que no pudieron ser transmitidos.

Las características generales del módulo I²C son las siguientes:

- Lógica independiente de maestro y esclavo.
- Soporte multi-maestro. No se pierden mensajes en el arbitraje.
- Detecta direcciones de dispositivos de 7 y 10 bits.
- Detecta llamados generales a direcciones como se define en el protocolo I²C.
- Modo repetidor de bus. Acepta todos los mensajes como esclavo sin importar la dirección.
- Ajuste automático de reloj SCL. Este provee demoras para que el procesador responda a un pedido de datos de un esclavo.
- Soporte para especificaciones de buses de 100 Khz. y 400 Khz.

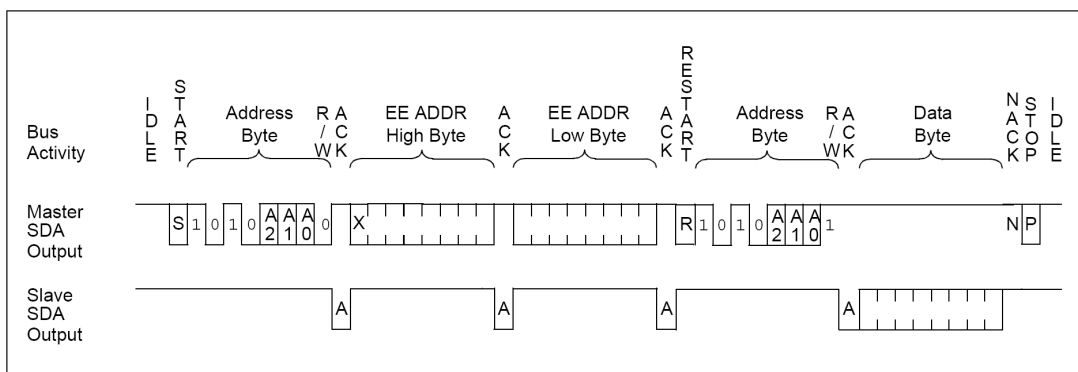


Figura 11. Mensaje I²C. Lectura de un byte de una memoria EEPROM

Módulo UART

El módulo de transmisión y recepción asincrónica universal o UART (Universal Asynchronous Receiver Transmitter) es uno de los módulos de entrada/salida serial disponible en los microcontroladores dsPIC. El UART es un sistema de comunicación asincrónica full-duplex (comunicación simultánea entre emisor y receptor) que permite comunicar al microcontrolador con otros dispositivos periféricos como computadoras personales, RS-232, RS 485, etc.

Las características principales del módulo UART son:

- Comunicación full-duplex de 8 o 9 bits.
- Opciones de paridad par, impar o ninguna.
- Uno o dos bits de parada.
- Generador integrado de velocidad de transmisión con prescaler de 16 bits.
- Velocidades de transmisión de 29 bps hasta 1,875 Mbps.
- Buffer de datos FIFO de transmisión y recepción de 4 niveles.
- Detección de errores de paridad, Framing y desbordamiento de buffers.
- Soporta interrupciones de transmisión y recepción.
- Soporte de retroalimentación (loopback) para diagnóstico.

En la figura X se muestra un diagrama simplificado de los bloques fundamentales del módulo UART. Estos bloques son tres:

- Generador para velocidad de comunicación.
- Transmisor asincrónico.
- Receptor asincrónico.

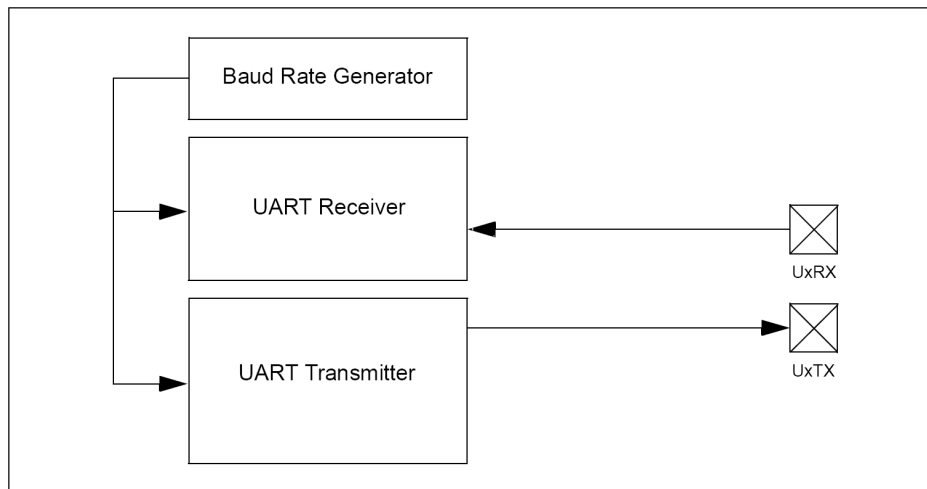


Figura 12. Diagrama simplificado de bloques del módulo UART.

CAN (Control Area Network).

El módulo controlador de red de área (CAN, Controller Area Network) es una interfase serie utilizado para la comunicación de otros dispositivos periféricos o microcontroladores. Este tipo de interfase o protocolo de comunicación fue diseñado para permitir que una red de dispositivos se comunique en entornos ruidosos.

Una diferencia importante con otros sistemas de comunicación es que CAN está orientado al mensaje y no al destinatario. Un dispositivo que realiza una transmisión tipifica un mensaje y los receptores lo filtran determinando si el tipo del mensaje es de su interés.

El módulo de comunicación CAN del dsPIC cuenta con registros de control, estado, transmisión, recepción, selección de velocidad de comunicación y registros especiales para el estado y control de interrupciones. La versión que implementa el módulo es la versión 2.0 del protocolo definida en la especificación BOSCH (su creador). Las principales características son:

- Longitud de datos entre 0 y 8 bytes.
- Velocidad de transmisión programable de hasta 1 Mbit por segundo.
- 6 buffers para filtros de aceptación: 2 asociados con mensajes de alta prioridad y 4 para baja prioridad.
- Fuente de reloj programable.
- Soporte de interrupción para transmisión, recepción y errores.

Para la comunicación entre todos los dispositivos de la red CAN el módulo provee las siguientes tipos de tramas:

- Trama de datos estándar: para transmisión normal de datos.
- Trama de datos extendida: para extender la transmisión normal de datos.
- Trama remota: es un pedido para que un nodo del sistema responda con la información solicitada.
- Trama de error: emitido por un nodo que detecta un error de transmisión.
- Espacio entre tramas: utilizado para separar varias tramas. Al separar 2 tramas los nodos receptores ganan tiempo para procesar la primera trama, ignorando el espaciado para luego procesar la segunda.

Temporizadores, Comparadores y Capturadores

Comparadores de Salida

El módulo comparador de salida tiene la capacidad de comparar un valor de un temporizador con el valor de uno o dos registros de comparación. Además permite la generación de un pulso simple de salida o un tren de pulsos de salida al producirse un evento de coincidencia en la comparación. Como la mayoría de los dispositivos periféricos del microcontrolador, es posible habilitar interrupciones para estas se generen al producirse un evento de coincidencia.

El dsPIC normalmente tiene alrededor de 8 canales de comparadores de salida, designados OCx, donde x es el número del comparador. Cada canal del comparador tiene los siguientes registros:

- OCxCON: registro de control de 16 bits.
- OCxR: registro de datos de 16 bits.
- OCxRS: registro de datos secundario de 16 bits.

Cada módulo comparador de salida puede trabajar de 3 modos diferentes:

- Coincidencia de comparación simple: al coincidir el temporizador y el registro de datos del comparador genera un pulso de salida y una interrupción. El registro de control se puede programar para generar el pulso:
 - Inicialmente bajo, luego pase a alto.
 - Inicialmente alto, luego pase a bajo.
 - Invertir el nivel del pulso.
- Coincidencia de comparación dual: al coincidir el temporizador y el registro de datos del comparador genera un nivel de salida (alto o bajo). Además compara el temporizador con el valor del registro de datos auxiliar, y si coincide invierte el nivel de salida.
- Modulación del ancho del pulso: en este modo se utiliza el registro de datos auxiliar con un valor de inicialización para el registro de datos. Cuando el temporizador coincide con el valor del registro de datos genera un nivel bajo en la salida y asigna al registro de datos el valor de inicialización. Cuando la cuenta del temporizador desborda (overflow), se genera un nivel alto en la salida.

Capturadores de Entrada

El módulo de captura de entradas es utilizado para capturar un valor de un temporizador cuando ocurre un evento en un pin de entrada (canal) del microcontrolador. Esta característica de captura de entradas es útil en aplicaciones que requieren medición de la frecuencia (período de tiempo) y de pulsos.

Cada pin de entrada puede ser configurado para capturar el valor de uno de dos temporizadores de 16 bits disponibles. La fuente de entrada utilizada como reloj de cada temporizador puede ser tanto un oscilador interno como externo.

El módulo de captura de entradas tiene varios modos de operación que pueden ser seleccionados a través de los bits de un registro de control denominado ICxCON. Los modos de operación disponibles son los siguientes:

- Captura del valor del temporizador cada 1 bajada de la señal en el pin de entrada.
- Captura del valor del temporizador cada 1 subida de la señal en el pin de entrada.
- Captura del valor del temporizador cada 4 subidas de la señal en el pin de entrada.
- Captura del valor del temporizador cada 16 subidas de la señal en el pin de entrada.
- Captura del valor del temporizador cada bajada y subida de la señal en el pin de entrada.

Este módulo de captura tiene un buffer FIFO de 4 niveles asociado a cada canal de entrada, lo que permite almacenar los valores que se van obteniendo en cada evento de captura. El usuario puede seleccionar la cantidad de eventos de captura necesarios para generar una interrupción a la CPU.

Temporizadores

Dependiendo de la variante específica, la familia del dsPIC ofrecen varios temporizadores de 16 bits.

Cada módulo temporizado se compone de los siguientes registros de 16 bits:

- TMRx: registro contador de 16 bits.
- PRx: registro de periodo asociado al temporizador.
- TxCON: registro de control de 16 bit asociado al temporizador.
 - (x es el número de temporizador asociado)

Cada temporizador tiene asociado bits para el control de interrupciones:

- 1 Bit Habilitación de interrupción.
- 1 bit de estado de interrupción.
- 3 bits de control para la prioridad de la interrupción.

Con algunas excepciones, todos los temporizadores de 16 bits tienen la misma funcionalidad. Es posible combinar dos temporizadores de 16 bits para formar uno de 32 bits. Los temporizadores se pueden clasificar en tres tipos diferentes de acuerdo con sus diferencias:

- Tipo A.
- Tipo B.
- Tipo C.

Los temporizadores Tipo A en general se utilizan, por sus características, en aplicaciones como reloj de tiempo real. Las características únicas sobre otros tipos de temporizadores son:

- Puede ser operado por un oscilador de baja potencia a 32 KHz.
- Puede ser operado de modo asincrónico por una fuente de reloj externa.

Los temporizadores Tipo B tienen las siguientes características sobre otros tipos de temporizadores son:

- Puede ser concatenado a un temporizador de Tipo C para formar un temporizador de 32 bits. Un bit del registro TxCON permite habilitar/deshabilitar esta característica.
- La sincronización de este tipo de temporizador se realiza luego de una lógica de pre-escalado, lo que permite obtener largos tiempos de temporización.

Los temporizadores Tipo C poseen las siguientes características sobre otros tipos de temporizadores:

- Puede ser concatenado a un temporizador de Tipo B para formar un temporizador de 32 bits.
- Los temporizadores de este tipo tienen la habilidad de disparar una conversión analógica/digital.

Conversores Analógico/Digitales

Los microcontroladores dsPIC, dependiendo de la familia y el modelo, pueden tener una gran cantidad de líneas de entrada(canales) para realizar conversiones analógico/digital (A/D). Normalmente cuentan con uno o dos módulos de conversión A/D de 16 canales. Cada modulo implementa un conversor de aproximaciones sucesivas (SAR). ****

Cada módulo puede tener un conversor una resolución de 10 bits o 12 bits de precisión.

Tanto los módulos de 10 bits como de 12 bits permiten disparar una secuencia de conversiones seleccionando un conjunto de canales para procesar

Las conversiones pueden generarse de las siguientes maneras:

- Manual: se genera cada vez que se lo indica
- Automática: se dispara de forma automática una nueva conversión al terminar la conversión en curso.
- Automática por evento: se dispara la conversión al producirse un evento asociado a un dispositivo que se configuró previamente. Ej: al terminar la cuenta de un temporizador.

En particular el módulo de conversión de 10 bits posee un módulo de reloj analógico que controla el período de conversión. Cada conversión se realiza en 12 ciclos de este reloj y puede tener como fuente al reloj de instrucciones (no disponible con CPU en modo SLEEP) o un reloj interno, denominado RC (disponible con CPU en SLEEP). El período de conversión puede ser alterado a través de 6 bits del registro de control ADCON3.

Otra característica de este módulo es que posee cuatro amplificadores S/H. La función de un amplificador S/H es retener la señal de un canal para luego procesarla por el conversor. Esto permite sincronizar la medición de las muestras de hasta cuatro canales para obtener los valores de conversión en un mismo instante de tiempo. A este tipo de muestreo se lo denomina “muestreo simultaneo”, mientras que el muestreo de canales separados se lo denomina “muestreo secuencial”.

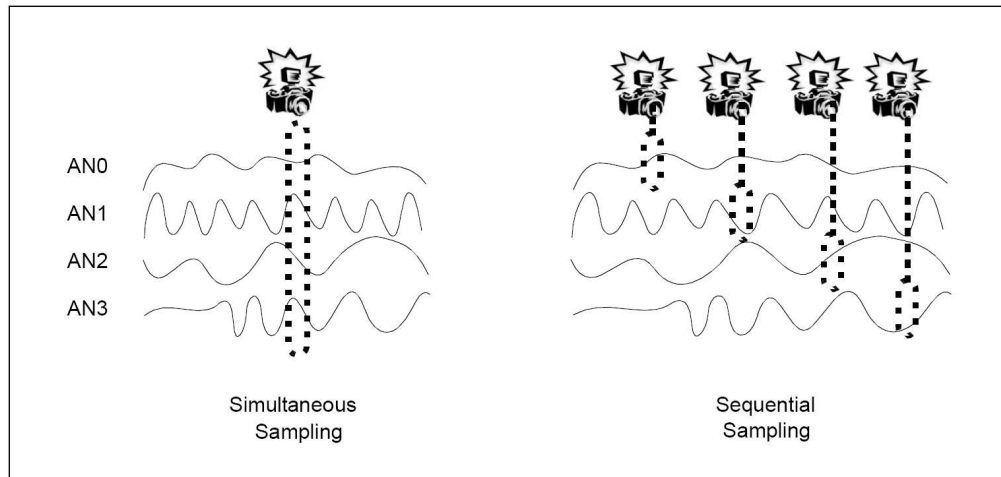


Figura 12. Muestreo Simultaneo vs. Secuencial.

Un módulo de 10 bits de resolución cuenta con :

- 16 entradas o canales para realizar conversiones.
- 2 entradas para voltaje de referencia.
- 4 amplificadores diferenciales S/H.
- 1 multiplexor para la selección de canales.
- 1 Controlador para obtener una secuencia de lecturas.
- 1 conversor A/D de 10 bits de resolución con 16 buffers de 16 bits.

Las características de un módulo de 10 bits de resolución son:

- Velocidades de conversión de hasta 1Msps (millón de muestras por segundo).
- Generación de interrupciones al finalizar conversiones.
- Programación de muestreo secuencia de canales.
- Muestreo simultaneo de hasta cuatro canales.
- Puede operar mientras la CPU esta en modo SLEEP o IDLE.

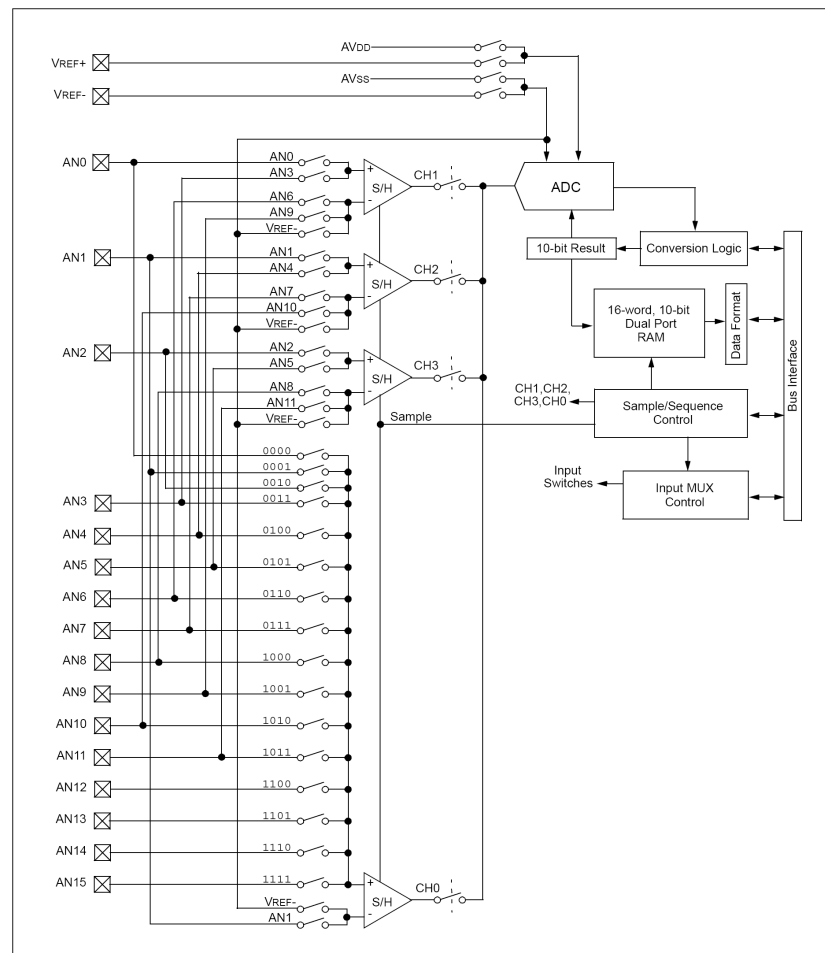


Figura 13. Diagrama de módulo de conversión A/D de 10 bits.

Un módulo de 12 bits de resolución cuenta con :

- 16 entradas o canales para realizar conversiones.
- 2 entradas para voltaje de referencia.
- 1 amplificador diferencial S/H.
- 1 multiplexor para la selección de canales.
- 1 Controlador para obtener un secuencia de lecturas.
- 1 conversor A/D de 12 bits de resolución con 16 buffers de 16 bits.

Las características de un módulo de 12 bits de resolución son:

- Velocidades de conversión de hasta 200Ksps (doscientas mil de muestras por segundo).
- Generación de interrupciones al finalizar conversiones.
- Programación de muestreo secuencia de canales.

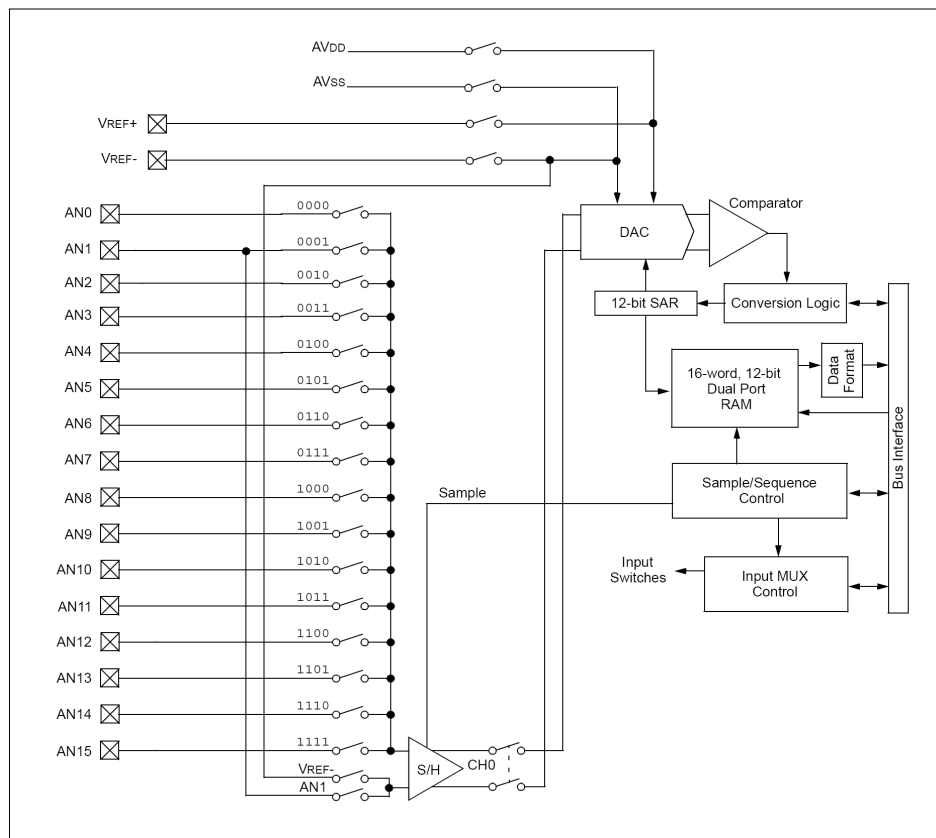


Figura 14. Diagrama de módulo de conversión A/D de 12 bits.

Codificadores de Cuadratura

Los codificadores de cuadratura (Quadrature encoders) son codificadores incrementales u ópticos utilizados en la detección de la posición y velocidad de rotación en sistemas de movimiento. Estos permiten seguir de cerca el giro de motores para controlarlos adecuadamente en muchos tipos de aplicaciones como motores de "cambio de reluctancia" y motores de inducción.

Un típico codificador incremental incluye una rueda con ranuras "atachada" al eje del motor y un módulo emisor/detector para sensar las ranuras de la rueda. Típicamente tienen tres salidas, denominadas Fase A, Fase B e índice, que pueden decodificarse para proveer información del movimiento del eje permitiendo determinar distancia y dirección.

Los dos canales denominados Fase A (QEA, Quadrature Encoder A) y Fase B (QEB, Quadrature Encoder B) están relacionados de manera que si el pulso de Fase A se adelanta al de Fase B la dirección de giro es determinada como positiva o hacia delante. Por el contrario, si el pulso de Fase B se adelanta al pulso de Fase A la dirección de giro es determinada como negativa o hacia atrás. El tercer canal denominado pulso índice se debe generar una vez por revolución y es utilizado como referencia para establecer la posición absoluta.

Las señales de cuadratura producidas por el codificador pueden tener cuatro estados diferentes. Estos estados son mostrados en un ciclo en la figura X. Notar que el orden de los estados se invierten al cambiar la dirección de giro.

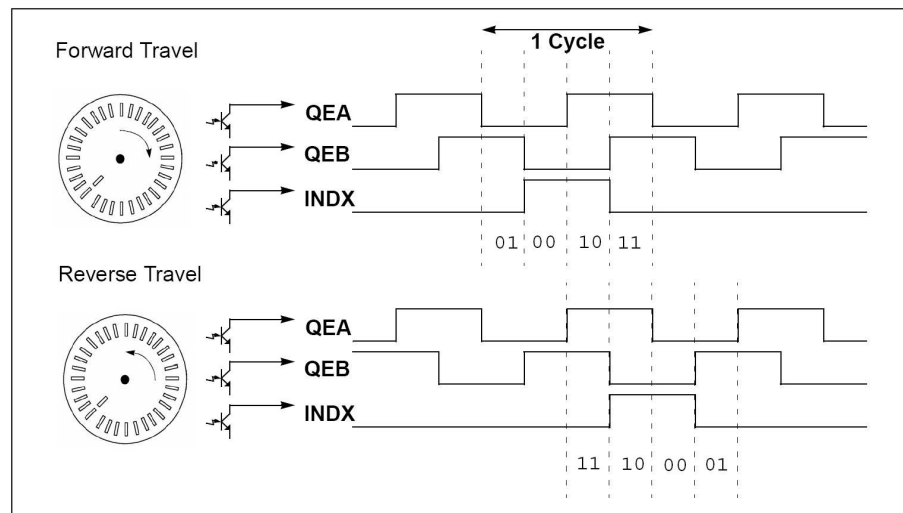


Figura 15. Señales de la interfase del codificador de cuadratura (QEI).

El codificador de cuadratura captura las señales de fase y el pulso índice, convirtiendo esta información en una cuenta numérica de los pulsos de posición. Generalmente la cuenta se incrementará cuando el eje del rotor rote en una dirección y decrementará cuando este rote en la dirección contraria.

La interfase del módulo de decodificación de cuadratura (QEI, Quadrature Encoder Interface) es una interfase para codificadores incrementales. El QEI consiste de una lógica para decodificación de cuadratura para interpretar las señales de Fase A y Fase B, y de un contador up/down para acumular la cuenta. Posee también un filtro digital para interferencias que acondicionan las señales de entrada.

Las características generales del módulo QEI incluye:

- 2 pines de entrada, dos para fase y uno para un pulso índice.
- Filtros digitales programables para las entradas.
- Contador de posición de 16 bits.
- Cuenta, con estado de dirección.
- Resolución de cuenta por dos (x2) y por cuatro (x4).
- Dos modos de reinicio de la cuenta de posición.
- Temporizador/contador de propósito general de 16 bits.
- Interrupciones generadas por QEI o por eventos del contador.

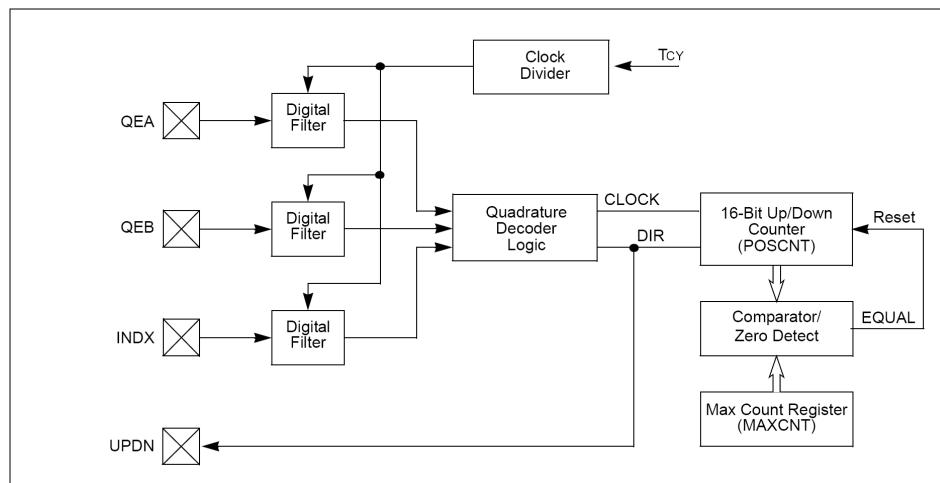


Figura 16. Módulo de interfase del codificador de cuadratura (diagrama simplificado).

El módulo QEI posee cuatro registros visibles para el usuario que se acceden de modo byte o word. Estos registros son:

- Registro de Estado/control (QEICON): este registro de 16 bits permite el control de operación y una indicación de las banderas de estado del estado del módulo.
- Registro de Control de Filtro Digital (DFLTCON): este registro de 8 bits permite el control de las operaciones sobre los filtros digitales de entrada.
- Registro de Control de Posición (POS CNT): este registro de 16 bits permite la lectura y escritura de la cuenta de la posición.
- Registro de cuenta Máxima (MAXCNT) : Este registro mantiene el valor que debe ser comparado con el registro de cuenta para algunas operaciones.

Compiladores, Simuladores y ambientes de desarrollo y bibliotecas

Microchip MPLAB

MPLAB IDE es un entorno de desarrollo gráfico (IDE) que permite editar, compilar y depurar programas de todos los tipos de microcontroladores de Microchip Technology (www.microchip.com) :

Este funciona entorno de desarrollo funciona bajo Windows y puede bajarse de manera totalmente gratuita desde el sitio de Microchip. Las herramientas que provee son las siguientes:

- administrador de proyectos.
- ventanas para inspección de registros, variables, memoria, llamados en pila, etc.
- compilador de assembler MPLAB ASM30 y linkeditor MPLAB link30.
- simulador de alta velocidad para CPU y periféricos.
- Inyector de estímulos y registro de salida para el simulador.

Microchip proporciona una versión de prueba por 60 días del entorno de desarrollo con su compilador de C, denominado MPLAB C30. Adicionalmente provee un conjunto de bibliotecas escritas en este lenguaje de programación que incluye conjuntos de funciones de:

- Matemática.
- Periféricos.
- Algoritmos DSP.
- Cancelación de eco acústico.
- Supresión de ruido.
- Reconocimiento de habla (en inglés).
- Encriptación simétrica y asimétrica.

La empresa CMX System (<http://www.cmx.com/>) ofrece varios tipos de sistemas operativos para realizar multitarea de procesos para Microprocesadores y Microcontroladores de varios fabricantes, entre ellos los dsPIC de Microchip. Tiene una versión para tiempo real denominado CMX-RTX de prueba disponible en su sitio Web. Este sistema operativo cuenta con las características:

- Soporte para interrupciones anidadas.
- Entrega de código fuente.
- Cambio de contextos muy rápidos.
- Bajo tiempo de latencia en interrupciones.
- Administración de:
 - Tareas: creación, ejecución, eliminación, etc.
 - Eventos: esperas y avisos de eventos.
 - Mensajes: enviar, recibir, esperar, etc.
 - Colas circulares.
 - Recursos: reserva, liberación, etc.
 - Semáforos.
 - Bloques de memoria.
 - UART.
- Completamente preemptivo.
- Puede administrar (scheduling) procesos de manera cooperativa o por lapso(slice) de tiempo.

MickroElektronika

MikroElektronika (<http://www.mikroelektronika.co.yu/spanish>) posee uno de los entornos de desarrollo integrado (IDE) mas cómodos, con asistente de código y resaltado de sintaxis. MikroElektronika ofrece entornos de desarrollo y compiladores para muchos microprocesadores y microcontroladores del mercado, entre ellos los dsPIC de Microchip. Otra característica interesante es que ofrece tres tipos de compiladores para diferentes lenguajes: Basic, Pascal y por supuesto C.

Entre las características del IDE y sus compiladores podemos mencionar:

- Explorador de código para explorar variables, funciones y procedimientos.
- Ayuda rápida con la lista de todas las rutinas disponibles.
- Asistente de Código y parámetros.
- Auto-corrector para errores comunes de tipeo.
- Resaltado de sintaxis.
- Templates de código.
- Depurador, puntos de ruptura(breakpoints) y ventana de inspección de variables.
- Estadísticas detalladas para uso de memoria (programas y datos) y de procedimientos.
- Compiladores para lenguajes Basic, Pascal y C.

Entre las bibliotecas que provee podemos mencionar:

- Trigonometría.
- Manejo de cadenas (strings).
- Conversiones.
- Utilidades generales.
- LCD.
- Teclado.
- Memoria Flash y EEPROM.
- Ethernet.
- Comunicación: CAN, CANSPI, PS/2, SPI, I²C, USB HID, USART, etc.

Con el entorno de desarrollo no se provee de un simulador, aunque provee de algunas herramientas que permiten el control y prueba de determinadas características del microcontrolador.

IAR Systems

IAR Systems (www.iar.com) provee un entorno de desarrollo integrado para construir y depurar aplicaciones embebidas. Como todos los demás fabricantes proveen compiladores para varios tipos de microprocesadores y microcontroladores, entre ellos los dsPIC. Provee una interfase común para assembler, compilador, administrador de proyectos, editor, compilador y depurador. El compilador para dsPIC utiliza C/C++. También implementa un simulador, llamado C-SPY, que soporta depuración para RTOs.

Las características mas importantes son:

- Editor de C/C++ con resaltado de sintaxis.
- Administrador de proyectos.
- Compilador ISO/ANSI C y C++ embebido con múltiples niveles de optimización de espacio y velocidad de ejecución.
- In-line assembler.
- Código mixto para Assembler y C/C++.
- Soporte para todas las características de DSP.
- El depurador C-SPY permite establecer breakpoints, simular I/O e interrupciones, visualizar registros, estructuras, variables, etc.

En cuanto a las bibliotecas provee:

- Todas las bibliotecas requeridas ISO/ANSI C: manipulación de caracteres, entrada/salida, utilidades generales, manejo de string, matemática y trigonometría, rutinas de bajo nivel, etc.
- Bibliotecas para manejo específico de todo el hardware para DSP.
- Extensiones de C y C++ con funciones matemáticas y de punto flotante.

HI-TECH Software

Hi-Tech Software (www.htsoft.com) ofrece entornos de desarrollo integrados y compiladores para una gran variedad de microprocesadores y microcontroladores (HI-Tide 3). En general provee tanto de versiones para Windows, Linux, MAC y Sun Solaris. Entre las características que ofrece se encuentran:

- Administración de proyectos.
- Depurador integrado permite breakpoints, visualización de variables y de memoria, etc.
- Editor construido en Eclipse. Provee resaltado de sintaxis, completa código, etc.
- Control de versiones.
- Compilador de ANSI C e integrable a otros IDE, como MPLAB.
- Código Assembler y C mixto.

Hi-Tech provee un sistema operativo de tiempo real, denominado SALVO, para chips con limitaciones severas de memoria RAM o ROM. Este sistema operativo proporciona multitarea y un uso eficiente de memoria. Todavía no esta disponible para las líneas dsPIC de Microchip.