Dedicatoria y Agradecimientos

Una página no es suficiente para agradecer a tantos(as) quienes nos han brindado su apoyo, colaboración, ayuda, afecto, cariño y sobre todo paciencia, para alcanzar esta meta, la cual significa el principio de una exitosa carrera.

Queremos dedicar este arduo trabajo, primero a Dios por ser nuestro guía y orientador en cada uno de nuestros pasos, haciendo posible que lográramos con éxito todas las metas que nos hemos trazado, no sólo en el desarrollo de este trabajo, sino en nuestras vidas.

Dedicamos también este trabajo a nuestros padres: Yurimar De Hurtado y Guillermo Alberto Hurtado (padres de Guillermo), así como a Ana C. Perdomo y José Caraballo (padres de Andrés) por estar siempre allí, en las buenas y malas, por ser nuestro pilar, nuestro apoyo incondicional; les agradecemos su guía, su ánimo y motivación para hacernos cada vez mejores personas y profesionales. A ustedes queremos dedicarles éste y todos los logros de nuestra vida.

Y nuestro profundo agradecimiento y reconocimiento a la Universidad Católica Andrés Bello y a nuestros profesores, por darnos una excelente formación profesional y la oportunidad de ser los pioneros de la carrera de Ingeniería de Telecomunicaciones en Venezuela, en especial a la Profesora María Cristi Stefanelli por sus orientación, su ayuda y apoyo en el desarrollo de este trabajo.

Agradecemos a Alejandro Teixeira por su amistad y la ayuda brindada durante el desarrollo de este Trabajo de Grado.

Agradecemos a Luís Guzmán por la confianza brindada y por la libertad para trabajar y desenvolvernos en la empresa Televen.

Resumen

El siguiente Trabajo Especial de Grado, consistió en el Diseño e Implementación de un Sistema de Telemetría que tiene como objetivo general el Monitoreo y Control remoto de las condiciones operativas de la estaciones repetidoras de Televen. Para tal fin se desarrollaron tres etapas fundamentales: El Software necesario para el Monitoreo y Control de las Estaciones Repetidoras, la Tarjeta de Adquisición y Control de Datos, y el Sistema de Transporte de Datos utilizando el sistema de Radio que actualmente está operativo en la Central. El Proyecto del Sistema de Telemetría mencionado se desarrolló para poder establecer una comunicación más directa entre la Central de Transmisiones y los equipos ubicados en las diferentes Estaciones Repetidoras; para ello se diseñó un Sistema de Monitoreo y Control basado en Visual Basic, con lo cual el operador ubicado en la central logra observar a través del monitor el estado de las Estaciones Repetidoras; la Tarjeta de Adquisición y Control de Datos creada con un PIC, recoge los parámetros del transmisor (voltajes, estado del transmisor, potencia Aural y Visual, temperatura de la sala, entre otros) y en caso de ser necesario permite al operador de la central manejar los distintos parámetros del transmisor de manera remota, ahorrando así el traslado de operadores hacia el transmisor y ofreciendo mayor comodidad y eficiencia. Se realizaron diversos diseños y pruebas las cuales lograron cumplir con el objetivo de comunicar las Estaciones Repetidoras con la Central de una manera sencilla, confiable y económica.

Palabras Claves: Telemetría, Monitoreo, Control, Estaciones Repetidoras, Central de Transmisiones.

Índice General

Dedicatoria y Agradecimientos i
Resumen ii
Introducción 1
Capítulo I 2
Planteamiento del Problema 2
I.1 Planteamiento del Problema 2
I.2 Objetivos 2
I.3 Limitaciones y Alcances 3
Capítulo II 5
Marco Teórico 5
II.1 Microcontrolador 6
II.2 Estudio del Microcontrolador PIC16F877 6
II.2.1.1 Diagrama de conexión del PIC 16F877 7
II.3 Estudio de las Estaciones Repetidoras 8
II.4 Parámetros a medir en cada Estación Repetidora 11
II.5 Radios UHF 12
II.6 Circuito Modulador 13
II.6.1 Modulación FSK 14
II.6.2 Modulador XR2206 15
II.6.3 Demodulador XR-2211 16
II.7 Componentes adaptadores de los parámetros a medir al PIC 17
II.7.1 Relé 18
II.7.2 Sensor de Temperatura 19
II.7.2.1 LM335 19
II.7.3 Detector de Audio y Video 20

Capítulo III 21
Marco Metodológico 21
III.1 Sistema de Telemetría Televen 21
III.2 Sistema de Monitoreo y Control 22
III.2.1 MsComm 22
III.2.2 Interfaz Final 24
III.2.3 Envío y Recepción de la Data 24
III.2.4 Estructura de la Trama 25
III.2.4.1 Descripción de la Data 26
III.2.5 Base de Datos 27
III.3 Tarjeta de Adquisición y Control de Datos 27
III.3.1 Microcontrolador 28
III.3.1.1 Pruebas realizadas con el PIC 16F877 28
III.3.1.2 Estructura de la programación del PIC 16F877 30
III.3.2 Circuito Conversor 32
III.3.2.1 Entradas Digitales 33
III.3.2.2 Entradas Analógicas 35
III.3.2.3 Salidas 37
III.3.3 Diseño de la Tarjeta de Adquisición y Control de Datos. 38
III.3.4 Pruebas de la Tarjeta de Adquisición y Control de Datos. 38
III.4 Sistema de Transporte de Datos 39
III.4.1 Radios UHF 40
III.4.2 Tarjeta de Transporte 43
III.4.2.1 Modulador FSK 43
III.4.2.2 Demodulador FSK 44
III.4.2.3 Circuito Conversor RS232-TTL. 46
III.4.2.4 Diseño de Tarjeta de la Estación Central y Repetidora 46
III.4.2.5 Pruebas Tarjetas de la Estación Central y Repetidora. 46
Capítulo IV 48
Resultados 48

IV.1 Sistema de Monitoreo y Control. 48
IV.1.1 Mscomm. 48
IV.1.2 Interfaz Final 51
IV.1.3 Interfaz de Transporte 54
IV.1.4 Interfaz Base de Datos. 58
IV.2 Tarjeta de Adquisición y Control de Datos. 59
IV.3 Sistema de Transporte 61
Conclusiones y Recomendaciones 66
Bibliografía 69
Apéndice A. Plano Electrónico Tarjeta de Adquisición y Control de Datos 71
Apéndice B. Plano Electrónico Tarjeta Modem de la Estación Central 73
Apéndice C. Plano Electrónico Tarjeta Modem de la Estación Repetidora 75
Apéndice D. Código Visual Basic 77
Apéndice E. Código Microcontrolador PIC16F877 84
Anexo A. Especificaciones Técnicas PIC16F877 89
Anexo B. Especificaciones Técnicas XR2206 93
Anexo C. Especificaciones Técnicas XR2211 97
Anexo D. Fuente de Poder Conmutada 98

Índice de tablas

Tabla 1. Estados y sus repetidoras 9
Tabla 2. Ubicación respecto a sus coordenadas de las Repetidoras 10
Tabla 3. Representación en ASCII de la data enviada. 49
Tabla 4. Representación en ASCII de la data recibida. 50

Índice de figuras

Figura 1. Diagrama de pines del PIC 16F877 7
Figura 2. Estados con estaciones repetidoras. 8
Figura 3. Estación de Caricuao 12
Figura 4. Radios UHF 13
Figura 5. Modulación FSK 14
Figura 6. Diagrama de Bloques del XR-2206 15
Figura 7. Diagrama de Bloques XR2211 16
Figura 8. Partes de un Relé de Armadura 18
Figura 9. a) Montaje de Calibración, b) Conexión de pines del LM335 19
Figura 10. Sistema de Telemetría implementado. 21
Figura 11. Ventana de trabajo Principal. 23
Figura 12. Circuito Activación PTT 25
Figura 13. Estructura de la Trama Enviada desde la Central. 26
Figura 14. Estructura de la Trama Enviada hacia la Central. 26
Figura 15. Esquema de Conexión del PIC16F877. 31
Figura 16. Conector DB-25 32
Figura 17. Circuito Adaptador 28V-5V 33

Figura 18 . Detector de Video 34
Figura 19. Detector de Audio 35
Figura 20. Sensor de temperatura LM335 36
Figura 21. Esquema externo de los filtros para Potencia Aural y Visual. 36
Figura 22. Rectificador para la Potencia Aural y Visual. 37
Figura 23. Parámetros de salida 37
Figura 24. Equipos de radio Motorola Radius: *(a)GM300, (b) M120* 40
Figura 25. Enlace Televen – El Cuño – Caricuao. 42
Figura 26. Línea de Vista Televen – EL Cuño. 43
Figura 27. Línea de Vista El Cuño- Caricuao. 43
Figura 28. Montaje Modulador FSK. 44
Figura 29. Montaje Demodulador FSK. 45
Figura 30. Circuito Conversor RS232-TTL 46
Figura 31. La trama enviada con recepción exitosa. 48
Figura 32. Visión principal del programa de Monitoreo y Control 52
Figura 33. Interfaz Estado Aragua 53
Figura 34. Interfaz Estado Nueva Esparta 53
Figura 35. Interfaz Estado Miranda 54
Figura 36. Ventana de Reporte 54
Figura 37. Ventana de Zona Restringida. 56
Figura 39. Impresión del Reporte 57
Figura 40. Guardar el Reporte 57
Figura 41. Base de Datos 58
Figura 42. Base de Datos hoja Excel 58
Figura 43. Gráficas de la Base de Datos 59
Figura 44. Circuito Impreso de la Tarjeta de Adquisición y Control de Datos. 60
Figura 45. Distribución de la Tarjeta de Adquisición y Control de Datos. 60
Figura 47. a) Señal Modulada; b) Señal Demodulada. 61
Figura 48. Circuito Impreso de la Tarjeta Estación Central. 62
Figura 49. Distribución de la Tarjeta Estación Central. 62

Figura 50. Acabado final de la Tarjeta de Transporte Central. 63
Figura 51. Circuito Impreso de la Tarjeta Estación Repetidora. 64
Figura 52. Distribución de la Tarjeta Estación Repetidora. 64
Figura 53. Tarjeta de Transporte Repetidora. 65
Figura 54. Acabado final del Hardware del Sistema de Telemetría. 65

Introducción

En virtud de establecer una comunicación más directa entre la Central de Transmisiones y los equipos transmisores ubicados en las diferentes Estaciones Repetidoras, TELEVEN se planteó la necesidad de implementar un Sistema de Telemetría que utilice como medio de transporte de datos el sistema de radio instalado actualmente y que se encargue del Monitoreo y Control remoto de las condiciones operativas de las Estaciones Repetidoras, para poder así, detectar fallas en los equipos y actuar rápidamente en caso de que alguna se presente.

Para cumplir con el objetivo planteado, se diseñó un Sistema que se divide en tres grandes bloques: El primer bloque consiste en una interfaz desarrollada bajo el ambiente de Windows utilizando Visual Basic 6.0 para poder hacer de la misma un ambiente de trabajo agradable y fácil de utilizar para el operador de la Central. El segundo bloque se basa en el desarrollo de una Tarjeta de Adquisición y Control de Datos utilizando un PIC que responde a la petición de reporte hecha por el operador, e interpreta parámetros de entrada, tales como: temperatura de la sala, estado del transmisor, Potencia Aural, Potencia Visual, encendido o apagado del Transmisor, entre otros. El tercer bloque es el diseño de un hardware confiable y económico que permita transportar los datos desde la Central hasta la Estación Repetidora y viceversa, utilizando los radios disponibles en cada una de las estaciones repetidoras.

El Sistema de Telemetría permite almacenar ciertos parámetros de las Estaciones Repetidoras al momento de solicitar un reporte, los cuales son utilizados para una base de datos que permite ver el comportamiento de éstos cada cierto período.

Capítulo I
Planteamiento del Problema

I.1 Planteamiento del Problema

Actualmente, Televen cuenta con un Sistema de Telemetría muy obsoleto y poco amigable que solo monitorea una sola Estación Repetidora ubicada en el interior del país. La interfaz sobre la cual trabaja esta basada en MS-DOS, por lo cual es muy complicado su manejo, y los equipos que utiliza para realizar las funciones de adquisición y modulación de datos ya han comenzado a presentar fallas.

Dadas las condiciones planteadas, la Vicepresidencia de Transmisiones de Televen se vio en la necesidad de solicitar una alternativa para el desarrollo de un Sistema de Telemetría que permita el Monitoreo y Control de ciertos parámetros de las Estaciones Repetidoras de forma sencilla, amigable, eficiente y rápida.

Es importante mencionar que el desarrollo de este Trabajo Especial de Grado, es la continuación de unas pasantías realizadas, de aquí que parte del Sistema ya se encontraba diseñado solo a la espera de ser complementado, modificado, examinado y culminado para su implementación en la empresa Televen.

I.2 Objetivos

Para establecer los objetivos Generales y Específicos que cubre el Trabajo Especial de Grado, es necesario tomar en cuenta que ya existen avances y objetivos cumplidos para solucionar el problema planteado:

General:

- Modificar, culminar e implementar un Sistema de Telemetría que permita establecer una comunicación más directa desde la Central de Transmisiones con los equipos (Transmisores de TV) ubicados en las diferentes Estaciones Repetidoras de Televen, de manera de poder monitorear y controlar con una noción mas clara el estado en que se encuentran dichos equipos.

Específicos:

- Complementar el Sistema de Monitoreo y Control bajo el ambiente de Windows en Visual Basic, para que ofrezca una interfaz amigable.
- Crear una Tarjeta de Adquisición y Control de Datos que permita almacenar los datos de la Estación Repetidora y actúe sobre ésta dependiendo de la orden enviada.
- Implementar un método económico y confiable que permita transportar los datos desde la Central hasta las Estaciones Repetidoras y viceversa.
- Crear una Base de Datos que permita observar el comportamiento que han presentado las estaciones repetidoras en cierto período.

I.3 Limitaciones y Alcances

Este Trabajo Especial de Grado incluye un Sistema de Monitoreo y Control bajo el ambiente de Windows basado en Visual Basic 6.0, el cual maneja los reportes dados por la estaciones repetidoras, dicho Sistema tiene todas las estaciones repetidoras de Televen en Venezuela; incluye una Tarjeta de Adquisición y Control de Datos para la Estación Repetidora Piloto, esta tarjeta se presenta en su circuito impreso lista para ser instalada; también incluye un método confiable y económico que permite transmitir los datos entre la central y las estaciones repetidoras, este

método se presenta en su circuito impreso listo para ser instalado en la Central y en la Estación Repetidora Piloto. El Trabajo Especial de Grado incluye el Diseño e Implementación de un Sistema de Telemetría a través de los elementos nombrados anteriormente.

Este Trabajo Especial de Grado no incluye la implementación del Sistema de Telemetría en todas las Estaciones Repetidoras de Televen, debido a su gran magnitud y complejidad se instaló en una Estación piloto, de manera que sea un Sistema que después de la realización de las diversas pruebas, correcciones, y recomendaciones ya cumplidas durante el periodo de la realización de TEG, se pueda implementar en todas las Estaciones Repetidoras de Televen a través de su mismo personal.

Capítulo II

Marco Teórico

El número, cada vez mayor, de puntos de experimentación y muestreo en las zonas de trabajo de campo, obligó a crear un sistema de adquisición de datos de bajo coste, fácilmente expandible, donde el monitoreo y control en tiempo real de los distintos sensores desde el propio centro fuera el principal condicionante, de esta necesidad surge los sistemas de Telemetría.

La Telemetría significa "medición a distancia" y es una tecnología que consiste en la medición de ciertos parámetros (velocidad, temperatura, aceleración, presión, voltajes, etc.) mediante unos sensores y el envío de esos datos "en directo" mediante ondas electromagnéticas, por ejemplo ondas de radio a un centro de recepción para su análisis, la telemetría permite observar y controlar distintos parámetros a distancia, lo cual lo hace una tecnología muy versátil y eficaz ya que ahorra tiempo, traslado, comodidad, etc.

La Telemetría se utiliza en infinidad de campos, desde la exploración científica (en naves no tripuladas como submarinos, aviones de reconocimiento y satélites) hasta diversos tipos de competición (por ejemplo, Fórmula 1 y MotoGP). (Bailey, David, 2002)

Para la realización de un Sistema de Telemetría es fundamental el conocimiento de ciertos aspectos que ayudan a crear las bases necesarias para la comprensión de todas las características y desarrollo general de este proyecto. A continuación se procede a describir estos aspectos:

II.1 Microcontrolador

Se denomina microcontrolador a un dispositivo programable capaz de realizar diferentes actividades que requieran del procesamiento de datos digitales y del control y comunicación digital de diferentes dispositivos; poseen una memoria interna que almacena dos tipos de datos; las instrucciones, que corresponden al programa que se ejecuta, y los registros, es decir, los datos que el usuario maneja, así como registros especiales para el control de las diferentes funciones del microcontrolador. En su interior poseen principalmente una ALU (Unidad Lógico Aritmética), memoria del programa, memoria de registros, y pines I/O (entrada y/0 salida). La ALU es la encargada de procesar los datos dependiendo de las instrucciones que se ejecuten, mientras que los pines son los que se encargan de comunicar al microcontrolador con el medio externo; la función de los pines puede ser de transmisión de datos, alimentación de corriente para l funcionamiento de este o pines de control especifico.

Los microcontroladores son muy utilizados para implementar controles automáticos. Por ejemplo, sensar la temperatura de un proceso, compararla con un valor almacenado en memoria y tomar la decisión de encender un equipo de calefacción si la temperatura baja de cierto valor, y además de ello mostrar el valor en un *display*. (De Rueda, Luís, 2003)

II.2 Estudio del Microcontrolador PIC16F877

Como preámbulo a la programación del PIC16F877 se procedió a realizar un intenso estudio y documentación acerca de su uso como microcontrolador. En tal investigación se abarcaron tópicos como: organización en memoria de los registros más importantes, manejo de las instrucciones básicas, configuración de la USART para la comunicación vía puerto serial, manejo de las interrupciones, módulo de conversión A/D para la medición de parámetros analógicos, entre otros.

Por su parte, como herramienta para la programación del PIC 16F877 se utilizó el programa MPLAB IDE v 7.21, el cual trabaja en lenguaje ensamblador y permite

configurar una extensa variedad de modelos de PICs para que realicen diversas tareas; además, fue necesario conseguir un programa que permitiese grabar el código realizado en MPLAB en el PIC, para lo cual se hizo uso del programa ICPROG v 1.5 que es un software gratuito y de un dispositivo especial para tal fin.

II.2.1.1 Diagrama de conexión del PIC 16F877

Para asegurar el óptimo funcionamiento del PIC fue necesario estudiar el diagrama de conexión del mismo, el cual viene dado por las funciones que tiene cada pin. El diagrama de los pines se muestra en la figura 1, donde se puede observar que el PIC requiere de una alimentación de 5 voltios (pines 1, 11 y 32) y de una puesta a tierra (pines 12 y 33), además de un oscilador externo constituido por un reloj de cuarzo de 4 MHz (entre los pines 13 y 14). Véase Anexo 1, datasheet del PIC16F877. (Microchip Inc, 2001)

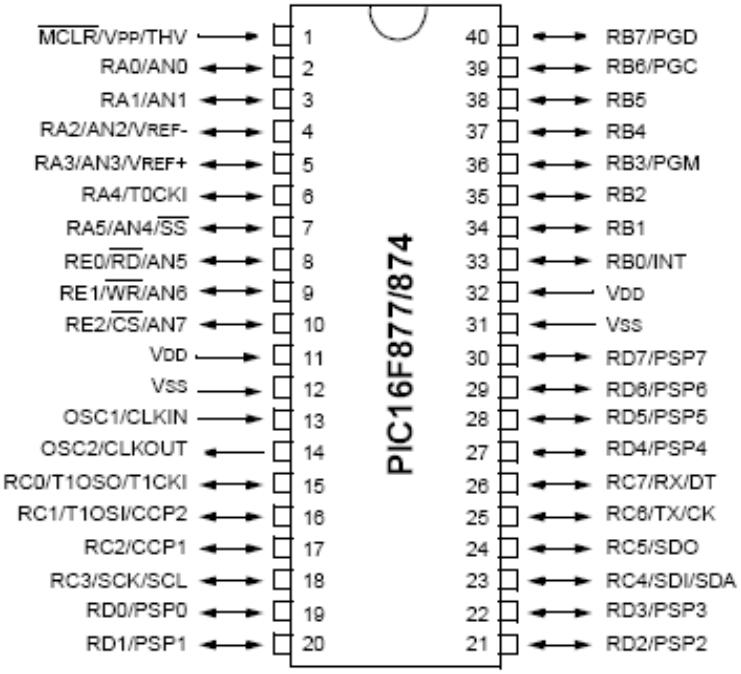


Figura 1. Diagrama de pines del PIC 16F877

II.3 Estudio de las Estaciones Repetidoras

Televen consta de un sistema de repetidoras las cuales se encuentran ubicadas a lo largo de toda Venezuela, de manera que le permita tener cobertura a nivel Nacional.

A continuación se presenta un mapa de los estados con Estaciones Repetidoras:

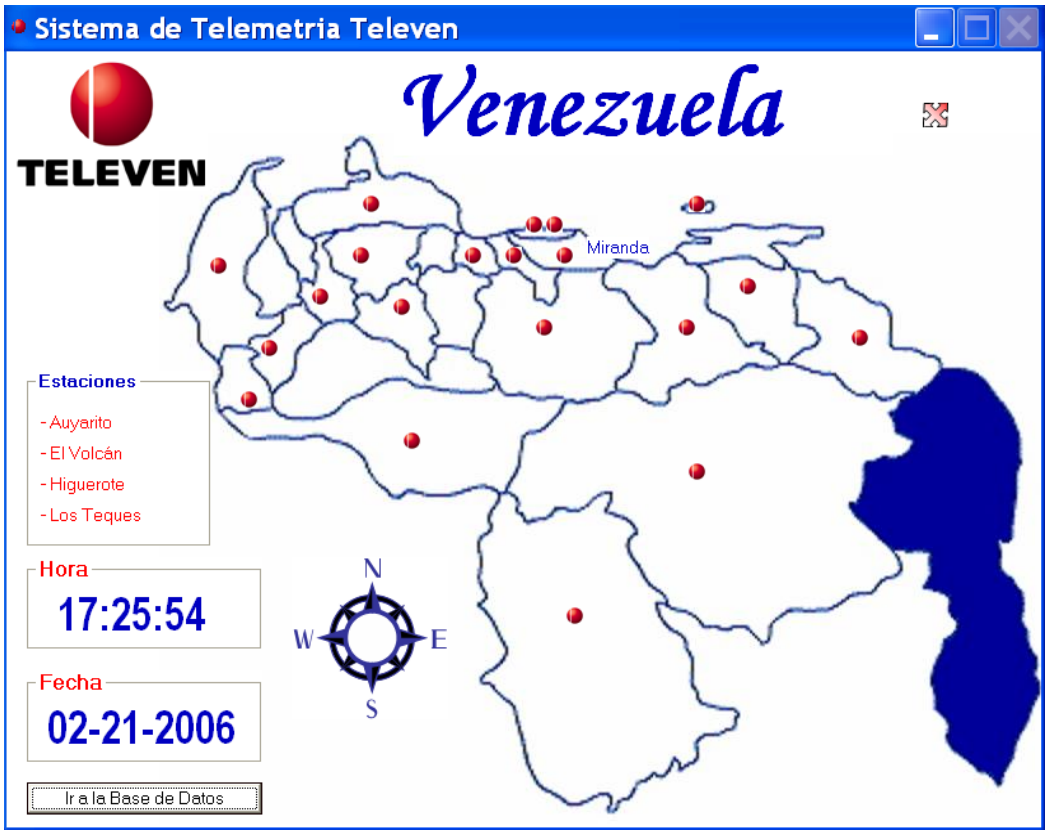


Figura 2. Estados con estaciones repetidoras.

Son 20 los estados que transmiten la señal de Televen y 34 las estaciones repetidoras distribuidas a nivel Nacional, en donde estados como Miranda poseen hasta 4 estaciones.

A continuación se presenta en la Tabla 1 los estados de Venezuela y cada una de las repetidoras pertinentes al mismo, a su vez se encuentra el ID con el que se identificará cada una de ellas.

Estado	Estación	ID
Amazonas	Puerto Ayacucho	**A**
Anzoátegui	El Tigre	**B**
	Puerto La Cruz	**C**
Apure	Achaguas	**D**
	San Fernando	**E**
Aragua	Maracay	**F**
Bolívar	Ciudad Bolívar	**G**
	Puerto Ordaz	**H**
Carabobo	Puerto Cabello	**I**
	Valencia	**J**
Delta Amacuro	Tucupita	**K**
Distrito Federal	Caricuao	**L**
	El Cuño	**M**
Falcón	Coro	**N**
Guárico	Platillón	**Ñ**
	Valle de La Pascua	**O**
Lara	Barquisimeto	**P**
	Carora	**Q**
Mérida	Mérida	**R**
Miranda	El Volcán	**S**
	Auyarito	**T**
	Higuerote	**U**
	Horizonte	**V**
	Los Teques	**W**
Monagas	Maturín	**X**
Nueva Esparta	Margarita	**Y**
Portuguesa	Araure	**Z**
Táchira	Ureña	**1**
	San Cristóbal	**2**
	San Telmo	**3**
Trujillo	Boconó	**4**
	Valera	**5**
Vargas	Litoral Central	**6**
Zulia	Maracaibo	**7**

Tabla 1. Estados y sus repetidoras

En la Tabla 2 se encuentra la latitud y longitud de cada una de las estaciones repetidoras de Venezuela por parte de Televen.

Estado	Estación	LATITUD			LONGITUD		
		Gr	Mi	Se	Gr	Mi	Se
Amazonas	Puerto Ayacucho	5	39	58	67	34	41
Anzoátegui	El Tigre	8	54	56	64	23	26
	Puerto La Cruz	10	10	28	64	38	5
Apure	Achaguas	7	40	56	68	13	34
	San Fernando	7	53	7	67	28	33
Aragua	Maracay	10	5	0	67	40	0
Bolívar	Ciudad Bolívar	8	7	50	63	32	37
	Puerto Ordaz	8	20	40	62	40	30
Carabobo	Puerto Cabello	10	29	42	65	5	1
	Valencia	10	12	7	67	59	38
Delta Amacuro	Tucupita	9	4	30	62	4	40
Distrito Federal	Caricuao	10	26	46	66	58	47
	El Cuño	10	31	12	66	52	50
Falcón	Coro	11	10	55	69	42	7
Guárico	Platillon	9	52	22	67	30	11
	Valle de La Pascua	9	13	34	65	19	54
Lara	Barquisimeto	9	59	8	69	16	23
	Carora	10	8	46	70	4	38
Mérida	Mérida	8	32	41	71	12	4
Miranda	El Volcán	10	25	4	66	51	4
	Auyarito	10	26	8	66	34	24
	Higuerote	10	34	48	66	3	10
	Horizonte	10	40	24	71	36	8
	Los Teques	10	22	0	66	58	20
Monagas	Maturín	9	44	17	63	11	53
Nueva Esparta	Margarita	10	59	49	63	50	47
Portuguesa	Araure	9	33	10	69	12	57
Táchira	Ureña	7	53	55	72	26	40
	San Cristóbal	7	52	19	72	14	10
	San Telmo	8	14	13	71	57	3
Trujillo	Boconó	9	14	30	70	11	8
	Valera	9	19	26	70	25	55
Vargas	Litoral Central	10	37	25	66	50	27
Zulia	Maracaibo	10	29	51	66	49	0

Tabla 2. Ubicación respecto a sus coordenadas de las Repetidoras

II.4 Parámetros a medir en cada Estación Repetidora

Cada una de las Estaciones Repetidoras maneja cierta cantidad de parámetros tanto digitales como analógicos, de los cuales se desea tener un monitoreo constante; y a su vez se operen salidas para el control de parámetros:

Entradas Digitales
- VA fuente DC. (28 volts Amplificadores). On/Off
- VB fuente DC (28 volts Amplificadores). On/Off
- VC fuente DC (28 volts Amplificadores). On/Off
- VD fuente DC (28 volts Driver). On/Off
- Estado del Transmisor. On/Off
- Reflejada. Hay o no.
- Presencia de Audio
- Presencia de Video

Entradas Analógicas
- Potencia Aural.
- Potencia Visual.
- Temperatura de la Sala.

Salidas
- Prender Transmisor
- Apagar Transmisor
- Modo Local
- Modo Remoto

Como se puede observar, la mayoría de los parámetros son digitales, valores que representan un evento específico y su negado, es decir, prendido o apagado, 0v y 5v;

las otras entradas son analógicas, las cuales deben ser representadas con valores numéricos para medir por ejemplo la Potencia Aural o Visual.

Además existen parámetros que se pueden controlar, como prender/apagar el transmisor, y cambiar el estado del Transmisor de Modo Remoto a Local o viceversa, este último es necesario ya que para prender/apagar el Transmisor es obligatorio que su estado sea Modo Local.

En la figura 3 se muestran varios de los equipos encontrados en la Estación Repetidora de Caricuao.

Figura 3. Estación de Caricuao

II.5 Radios UHF

Actualmente los equipos de Radio UHF mantienen comunicación con las estaciones repetidoras ubicadas tanto en Dto. Federal como en el Edo. Miranda; su uso es debido a que en muchas de las repetidoras no existe señal telefónica, siendo

necesarios para la comunicación con el personal técnico que se encuentre en algún momento en la repetidora.

En la figura 4 se encuentra la foto de los radios UHF ubicados en la Central, que se utilizan para el transporte de la data del Sistema de Telemetría.

Figura 4. Radios UHF

II.6 Circuito Modulador

El desarrollo de dicho Hardware durante el periodo de pasantías fue un Circuito Modulador OOK, debido a que las ventajas que ofrece dicha Modulación no son las mas óptimas para la transmisión de señales vía radio, se llegó a la conclusión de realizar una modificación en la fase del Trabajo Especial de Grado. Esta modificación se realizó debido a las sustanciales mejoras que muestra la Modulación FSK con respecto a la Modulación OOK en la inmunidad al ruido e interferencias.

II.6.1 Modulación FSK

FSK (Frequency-shift keying), es una modulación de frecuencia donde la señal moduladora (datos) es digital. Los dos valores binarios se representan con dos frecuencias diferentes (f_1 y f_2) próximas a la frecuencia de la señal portadora f_p.

$$v(t) = \begin{cases} V_p \ sen(2\pi \ f_1 \ t) \text{ para un "1" binario} \\ V_p \ sen(2\pi \ f_2 \ t) \text{ para un "0" binario} \end{cases} \qquad (2.1)$$

Generalmente f_1 y f_2 corresponden a desplazamientos de igual magnitud pero en sentidos opuestos de la frecuencia de la señal portadora, figura 5.

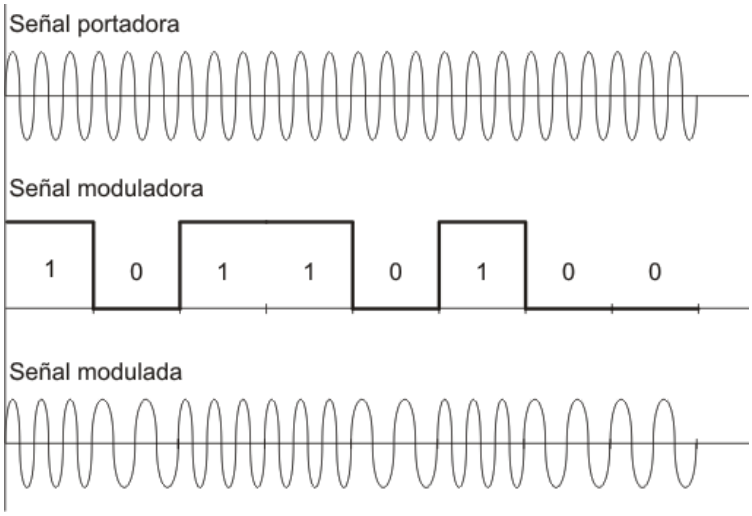


Figura 5. Modulación FSK

Para tal fin, fue necesario diseñar un circuito modulador FSK con la ayuda de los chips XR2206 (Modulador) y XR2211 (Demodulador). A continuación se presentará una descripción de cada uno de estos chips.

II.6.2 Modulador XR2206

El XR-2206 es un circuito integrado monolítico que trabaja como un generador de funciones, el XR-2206 es capaz de producir en alta calidad y con gran exactitud ondas senoidales, cuadradas, triangulares, rampa, etc. A la salida del circuito las formas de onda pueden variar su amplitud por medio de un voltaje externo. La frecuencia de operación del circuito integrado se puede seleccionar externamente, y tiene una gama de frecuencias que va desde 0.01 Hz hasta 1MHz. El circuito integrado XR-2206 es ideal para las comunicaciones, instrumentación que requiera de un tono senoidal o de la generación de una señal FSK (Frecuency Sfhit-Keying). A continuación se presenta el diagrama de bloque del circuito integrado XR-2206. En el Anexo B, se incluye parte del datasheet de este integrado para complementar la información. (Exar Corporation, 1997)

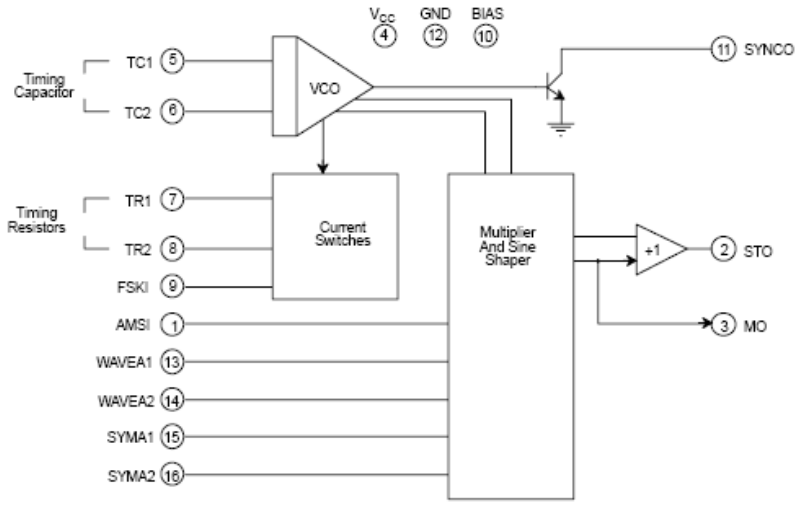


Figura 6. Diagrama de Bloques del XR-2206

El XR-2206 abarca cuatro bloques funcionales:

1. Oscilador voltaje-controlado (VCO).

2. Multiplicador y Sine-Shaper.
3. Buffer Amplificador de Ganancia Unitaria.
4. Un Sistema de interruptores.

El VCO produce una frecuencia a la salida proporcional a la corriente de la entrada, que es fijada por una resistencia en los terminales de sincronización.

II.6.3 Demodulador XR-2211

El XR-2211 es un sistema monolítico de fase-bloqueado del lazo (PLL) diseñado especialmente para los usos de comunicaciones de datos. Es utilizado comúnmente como un módem del FSK. El XR2211 posee una ancha gama de frecuencias las cuales varían desde 0.01 Hz. Hasta 300 KHz. El circuito consiste en un PLL básico para seguir la señal de entrada dentro del ancho de banda, posee un detector de fase que se utiliza para detectar la portadora y también tiene un comparador de voltaje FSK que permite realizar dicha demodulación. Los componentes externos del circuito se utilizan para fijar independientemente la frecuencia central, el ancho de banda, etc. (Exar Corporation,1997)

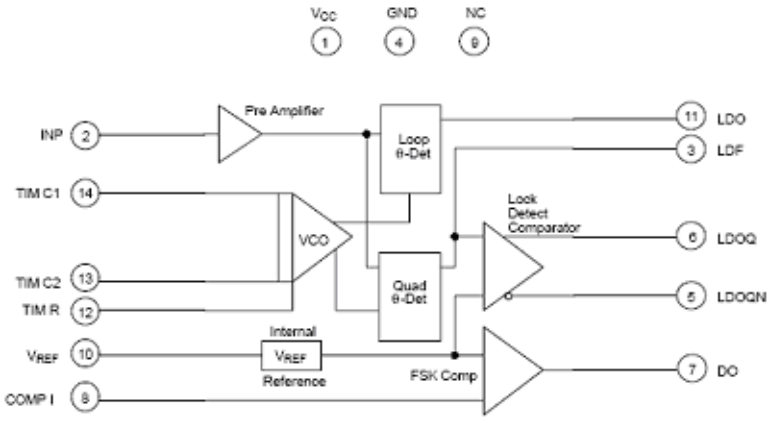


Figura 7. Diagrama de Bloques XR2211

El PLL principal que está dentro del XR-2211 esta constituido por un preamplificador a la entrada, un multiplicador análogo usado como detector de fase y un oscilador controlador de voltaje VCO). El preamplificador se utiliza como limitador, tal que las señales de entrada sobre 10mV rms son amplificadas a una señal constante de mayor nivel. El multiplicador - detector de fase actúa como una compuerta digital. Su salida (sin filtro) produce la suma o resta de las frecuencias de la entrada y de la salida de VCO. El VCO es un oscilador controlado donde su corriente normal de entrada (fO) se fija por un resistor (R0) a tierra y es llevada al detector de fase a través de R1. La salida del detector de fase produce la suma y la diferencia entre la frecuencia de entrada y la frecuencia de salida del VCO (conectadas internamente). El comparador del FSK se utiliza para determinar si el VCO está sobre o por debajo de la frecuencia central (comparador del FSK). Esto producirá salidas activas bajas y altas dependiendo de la entrada. En el Anexo C se incluye parte del datasheet del integrado para una información mas completa.

II.7 Componentes adaptadores de los parámetros a medir al PIC

Como se ha mencionado anteriormente, existen parámetros en las estaciones repetidoras (voltajes, estado del transmisor, temperatura de la sala, entre otros) de los que se necesita saber su estado en la Central de Televen, y para que estos parámetros puedan ser interpretados correctamente por la Tarjeta de Adquisición y Control de Datos (PIC) es fundamental realizar ciertas conversiones o modificaciones en cada uno de ellos, dependiendo de las características particulares que posean.

El componente que servirá de mayor ayuda para estas conversiones es el relé, seguidamente será necesario un sensor de temperatura, un circuito detector de audio, un circuito detector de video, entre otros. A continuación se ofrece una breve descripción teórica de cada uno de ellos, para luego ampliar su desarrollo en la fase metodológica correspondiente.

II.7.1 Relé

El Relé es un interruptor operado magnéticamente. Este se activa o desactiva (dependiendo de la conexión) cuando el electroimán (que forma parte del Relé) es energizado (le damos tensión para que funcione). Esta operación causa que exista conexión o no, entre dos o más terminales del dispositivo (el Relé), figura 8.

Esta conexión se logra con la atracción o repulsión de un pequeño brazo, llamado armadura, por el electroimán. Este pequeño brazo conecta o desconecta los terminales antes mencionados.

- **Ventajas del Relé:**

- Permite el control de un dispositivo a distancia. No se necesita estar junto al dispositivo para hacerlo funcionar.

- El Relé es activado con poca corriente, sin embargo puede activar grandes máquinas que consumen gran cantidad de corriente.

- Con una sola señal de control, puedo controlar varios Relés a la vez.

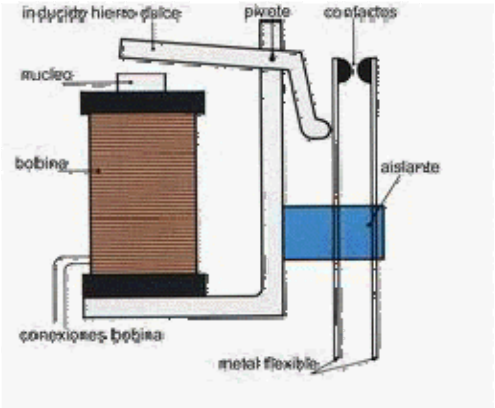


Figura 8. Partes de un Relé de Armadura

II.7.2 Sensor de Temperatura

El sensor que servirá de ayuda para obtener la medición de la temperatura de la sala es el componente electrónico LM335. A continuación se muestra una breve descripción sobre dicho componente.

II.7.2.1 LM335

El LM335 es un sensor de temperatura de precisión que puede ser calibrado fácilmente. Su voltaje de interrupción es directamente proporcional a la temperatura absoluta en 10mV/oK. El circuito tiene una impedancia dinámica de menos que 1W y funciona dentro de una gama de la corriente desde 450mA hasta 5mA. Calibrado en +25oC, El LM335 tienen un error típico de menos que 1ºC sobre una gama de temperaturas 100ºC. (National Semiconductor, 2000)

Para utilizar el LM335 como sensor de temperatura es necesario calibrarlo primero, hay un método fácil para calibrar el dispositivo para exactitudes más altas, figura 9.

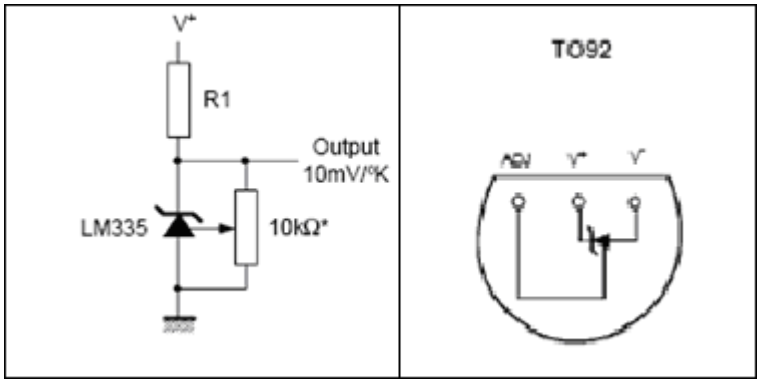


Figura 9. a) Montaje de Calibración, b) Conexión de pines del LM335

II.7.3 Detector de Audio y Video

La señal de audio y de video son dos mas de los parámetros a tomar en cuenta para su monitoreo. Para lograr detectar tanto la presencia de audio como la de video es necesario el diseño de un circuito que logre identificar su presencia, y a la vez activar un relé para dar a conocer el estado en que se encuentra cualquiera de las dos señales.

Una mayor explicación sobre el diseño y funcionamiento sobre lo descrito se encuentra en la fase metodológica correspondiente a la Tarjeta de Adquisición y Control de Datos.

Capítulo III

Marco Metodológico

Debido a la gran magnitud del proyecto se optó por dividirlo en 3 fases; la primera el Sistema de Monitoreo y Control, la segunda, la Tarjeta de Adquisición y Control de Datos y la tercera fase es el Sistema de Transporte de Datos desde la Estación Central hasta la Estación Repetidora y viceversa.

Antes de adentrar en estas tres fases se explicará a grandes rasgos el Sistema de Telemetría implementado en Televen.

III.1 Sistema de Telemetría Televen

Existen diversos sistemas de telemetría que permiten controlar y monitorear los datos en tiempo real; a continuación en la figura 10 se presenta en grandes rasgos el Sistema de Telemetría diseñado e implementado en el Trabajo Especial de Grado.

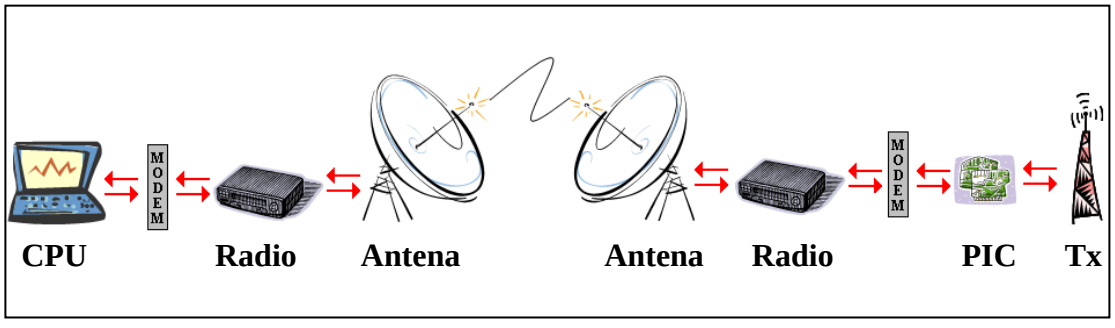


Figura 10. Sistema de Telemetría implementado.

Como se muestra en la figura el Sistema de Telemetría consta de un CPU el cual se encuentra ubicado en la Estación Central (Caracas), dicho CPU contiene el

Sistema de Monitoreo y Control realizado en Visual Basic 6.0; del CPU se envía la información con la finalidad de solicitar un reporte de alguna Estación Repetidora en específico, esta información es remitida al MODEM, luego de modulada la señal es enviada vía Radio hacia todas las estaciones repetidoras, cada una de estas estaciones comparan la data enviada de manera de ver si es a ella a quien se le solicitó el reporte, solo una estación identificará la data como propia a través del código único de cada una de las estaciones, dicha información es procesada por el PIC o Tarjeta de Adquisición y Control de Datos, la cual obtiene la información de los diferentes parámetros a medir y la envía de vuelta al radio y este a su vez a la Estación Central que se encargara de procesar dichos datos.

III.2 Sistema de Monitoreo y Control

Para llevar a cabo el desarrollo de este bloque del Sistema de Telemetría fue necesario diseñar dos aplicaciones distintas basadas en el mismo principio, una especialmente diseñada para la realización de pruebas y otra específicamente diseñada para el usuario final, llamadas Mscomm e Interfaz Final respectivamente.

Para el Sistema de Monitoreo y Control se utilizó el programa Visual Basic 6.0, ver en el Apéndice D el código referente a una estación repetidora, el cual permite manejar los puertos de la computadora de manera práctica y eficiente, además ofrece una interfaz amigable al operador. El Sistema de Monitoreo y Control es de suma importancia ya que es la interfaz inicial y final de todo el Sistema de Telemetría.

III.2.1 MsComm

Para la realización de pruebas fue necesaria la modificación de una pequeña aplicación para el manejo del puerto serial adaptándola a los requerimientos del sistema a implementar, ya que era necesario transmitir cadenas de datos continuos,

cálculo de la cantidad de bits errados en una cadena de caracteres (BE→ Bit error), conversiones ASCII y muy importante trabajo de datos a nivel de bit para así poder realizar pruebas y obtener resultados que permitieran el desarrollo de la aplicación final. El MsComm tiene la interfaz mostrada en la figura 11:

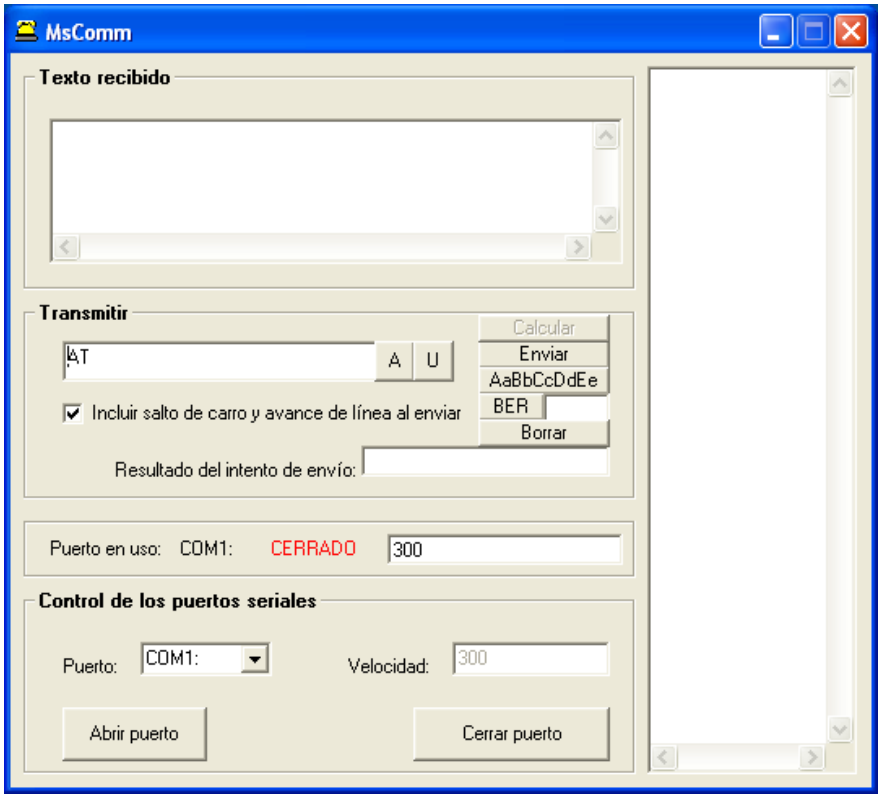


Figura 11. Ventana de trabajo Principal.

Mediante dicha interfaz es posible cambiar a conveniencia el puerto de comunicaciones a usar, la velocidad de transmisión y recepción, además de que permite transmitir largas y continuas cadenas de caracteres, así como la decodificación ASCII de los caracteres recibidos para poder descomponerlo en su forma binaria. Cada estación está identificada con un código que por motivos de prueba es un carácter de 8 bits, como por ejemplo la letra "A", que fue designada para la Estación Piloto, por lo que la Tarjeta de Adquisición y Control de Datos ubicada en esta estación sólo responderá cuando reciba este caracter.

III.2.2 Interfaz Final

Para que la interfaz final fuese amigable al operador se colocó un mapa de Venezuela donde se ubicó varios logotipos de Televen en el espacio que representa a los estados de Venezuela, exceptuando de éstos los estados que no poseen estaciones repetidoras para transmitir la señal de Televen, de manera que haciendo click sobre el logo se muestra el mapa de cada uno de los estados con las repetidoras correspondientes a cada uno de ellos, y al hacer click sobre una repetidora se envía automáticamente una petición de reporte a través de una cadena de caracteres específica que identifica a la estación repetidora, de manera que sólo ella responda a la petición del reporte.

El Sistema de Monitoreo y Control se basa en el envío y recepción de data a través del puerto serial RS-232. A continuación se explica ambas etapas del sistema.

III.2.3 Envío y Recepción de la Data

Para el envío de la data el Sistema abre el puerto serial por un lapso de 2 seg, en este lapso de tiempo se procede a realizar el PTT (Push to Talk) del radio, para ello se utiliza uno de los pines del puerto de comunicaciones (DTR) que se activa a 12V a conveniencia vía software y un hardware adicional que funciona en paralelo a éste, el cual permite usar dicha señal para activar el radio. Para su implementación se utilizó un transistor cuya función es la de un suiche, es decir, cuando el nivel del pin DTR de la interfaz EIA-RS232 sube a +12 voltios, a la salida de dicho transistor hay 0 voltios por lo que se polariza la bobina del relé con una fuente externa de +12 voltios y éste cambia al estado de normalmente cerrado, activando el PTT del radio. El diagrama circuital de este hardware se muestra en la figura 12.

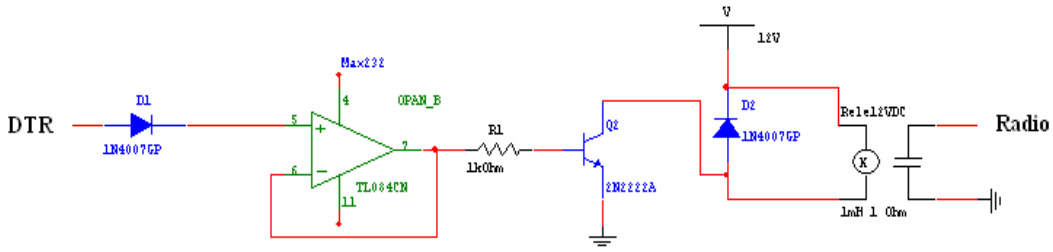


Figura 12. Circuito Activación PTT

Al instante de realizar el PTT, se procede a transmitir la data por el puerto serial, esta data esta conformada por una cadena de caracteres de 5 bytes, la cual es transmitida a una velocidad de 300 baudios/seg, la duración del PTT es de 500 mseg. Al culminar esos 500 mseg., se activa un timer el cual tiene como función chequear cada 100 mseg si el puerto serial a recibido el reporte; si el puerto serial no recibe el reporte en 2 seg, el Sistema mostrará en pantalla una Advertencia de que el reporte ha sido fallido, en cambio si el puerto serial recibe el reporte, mostrará en pantalla la información de que el reporte ha sido completado con éxito, en ambos casos el Sistema procede a cerrar el puerto serial.

III.2.4 Estructura de la Trama

El Sistema de Monitoreo y Control transmite a través del puerto serial una trama específica, la cual consta de 5 bytes (40 bits), y estructurada de la siguiente forma:

- **Bit de Inicio:** Se utiliza para el sincronismo de los modem.
- **ID:** Contiene la dirección de la estación a la cual va dirigida la data.
- **Data:** Determina la orden a realizar por el PIC.
- **Checksum:** Contiene la suma del ID con la Data enviada para verificar la correcta transmisión.
- **Bit de Parada.** Se utiliza para notificar la culminación de la trama.

A continuación se observa en la figura 13 la estructura de la trama:

Bits de Inicio (8)	ID (8)	Data (8)	Checksum (8)	Bits de Parada (8)

Figura 13. Estructura de la Trama Enviada desde la Central.

Así mismo, el PIC envía una trama conformada por 11 bytes (88 bits), la cual esta estructurada de la siguiente forma:

- **Bit de Inicio:** Se utiliza para el sincronismo de los modem.
- **ID:** Contiene la dirección de la estación
- **Data:** Contiene la información del estado de los parámetros analógicos y digitales.
- **Checksum:** Contiene la suma del ID con la Data enviada para verificar la correcta transmisión.
- **Bit de Parada.** Se utiliza para notificar la culminación de la trama.

A continuación se observa en la figura 14 la estructura de la trama:

Bits de Inicio (8)	ID (8)	Data (64)	Checksum (8)	Bits de Parada (8)

Figura 14. Estructura de la Trama Enviada hacia la Central.

III.2.4.1 Descripción de la Data

- **Data enviada desde la Central**

La Data enviada desde la central es fundamentalmente para realizar tres órdenes, en donde cada una tiene una configuración de 8 bits distinta:

Reporte - Encendido del Transmisor - Apagado del Transmisor.

- **Data enviada desde la Estación Repetidora**

Contiene la información de los parámetros, la cual se encuentra distribuida de la siguiente manera: 16 bits para el estado de los parámetros digitales, y 2 bytes por cada uno de los parámetros analógicos (3 entradas analógicas: Temperatura de la Sala, Potencia Aural y Visual).

III.2.5 Base de Datos

El Sistema de Monitoreo y Control posee una función que permite acceder a la Base de Datos, en donde se tienen archivados los reportes guardados de cada estación. La Base de Datos fue realizada bajo el ambiente de Excel, para ello se crearon 34 libros de Excel donde cada libro representa una estación de Televen; dichos libros contienen 12 hojas, cada hoja representa un mes del año, de manera que al guardar el reporte el sistema revisa la estación y la fecha del reporte y lo ubica automáticamente en el libro y hoja que corresponde.

La Base de Datos tiene la capacidad de almacenar 31 reportes mensuales, equivalente a un reporte diario, aunque se puede aumentar a la capacidad que el usuario decida. Conjuntamente la Base de Datos posee la facultad de crear gráficas dinámicas, es decir; que dichas gráficas son cargadas a medida que los reportes son guardados. Cabe destacar que las gráficas son solo en base a los parámetros analógicos (Potencia Aural, Potencia Visual y Temperatura), ya que son éstos los que tienden a tener un comportamiento ligeramente variable.

III.3 Tarjeta de Adquisición y Control de Datos

La Tarjeta de Adquisición y Control de Datos consiste básicamente en un microcontrolador PIC 16F877 y un circuito conversor para realizar las tareas de conversión de voltajes y mantener la compatibilidad entre el microcontrolador y el

transmisor. Cada una de las etapas que constituyeron la programación e implementación de la Tarjeta de Adquisición y Control de Datos será explicada a continuación, así como la del circuito conversor.

III.3.1 Microcontrolador

III.3.1.1 Pruebas realizadas con el PIC 16F877

Luego de la documentación e investigación hecha se procedió a realizar varias pruebas que sirven de gran ayuda para la implementación de la Tarjeta de Adquisición y Control de Datos. La descripción de dichas pruebas se hace a continuación:

- **Manejo de las Entradas y Salidas del PIC**

La primera prueba realizada consistió en programar dos de los pines del puerto A como entradas digitales (en los cuales se conectaron 2 interruptores) y dos pines del puerto B como salidas también digitales (en los cuales se colocaron 2 leds) de manera que cuando el valor lógico de las entradas fuera cero, en las salidas del PIC se encendieran los leds.

- **Comunicación vía serial con la PC**

Se programó el PIC de manera que se comunicara con el programa de prueba utilizado (Mscomm) que corría en el CPU. La prueba realizada consistió en que el PIC esperase recibir un caracter cualquiera, lo visualizara en forma binaria sobre leds y lo volviese a transmitir a la PC a modo de eco, de manera de que el caracter enviado pueda verse en el programa Mscomm. La recepción del caracter por parte del PIC se controló mediante la habilitación de las interrupciones periféricas.

- **Comparación de caracteres**

Se programó el PIC de manera que comparase el caracter enviado por la PC con un caracter específico, por ejemplo letra "A". Básicamente, si el caracter enviado desde la PC es una "A", el PIC envía de vuelta a la PC una "B", cuya representación binaria corresponde a los niveles de voltaje conectados a las entradas del puerto B y en caso de que la letra enviada desde la PC sea diferente a la "A", el PIC no responde.

- **Conversión A/D**

La prueba consistió en establecer uno de los pines del PIC como entrada analógica, a la que se le conectó un divisor de voltaje para poder variar el voltaje de entrada de ese pin y asegurar el correcto funcionamiento del módulo. Dicho voltaje analógico entra al pin y el PIC lo convierte a digital usando una representación de 10 bits y lo almacena en dos registros (ADRESH – ADRESL) que el PIC tiene reservados para tal fin.

Como los registros que maneja el PIC 16F877 son de 8 bits y los bits útiles de la conversión son sólo 10, era necesario realizar una serie de cálculos que permitiera visualizar el valor analógico de la entrada en pantalla. Para tal fin, se utilizaron las siguientes fórmulas:

$$\text{Resolución} = \left(V_{ref}{}^{+} - V_{ref}{}^{-}\right)/1024 \qquad (2.2)$$

$$V_{\text{analógico}} = \text{Valor decimal de los 10 bits} * \text{Resolución} \qquad (2.3)$$

Por ejemplo si el $Vref^{+}$ = 5 Voltios y el $Vref^{-}$ = 0 Voltios, la resolución es de 5V/1024 = 4,8828 mVoltios/bit, por lo cual si el resultado de 10 bits almacenado es 0100110010 (306), el voltaje analógico que entró al PIC es de 306 * 4,88 mVoltios/bit = 1,4941 Voltios.

Luego de realizar todas las pruebas mencionadas anteriormente, se procedió a realizar un sólo programa que ejecutara en conjunto todas las acciones necesarias para el control y monitoreo de la estación mediante el PIC.

III.3.1.2 Estructura de la programación del PIC 16F877

El código desarrollado para la programación del PIC se muestra en el Apéndice E y se realizó en el software MPLAB, siguiendo una estructura que se basa en los pasos que se describen a continuación:

1. **Rutina de inicialización:** Proporciona al PIC la configuración con la cual debe trabajar, es decir, se especifican los puertos que funcionan como salidas y los que funcionan como entrada, en el caso de las entradas se especifican cuales son digitales y cuales son analógicas.

2. **Habilitación de interrupciones:** Se habilitan las interrupciones relacionadas con la transmisión y recepción serial mediante las cuales se activa una bandera que produce la ejecución de cierta rutina una vez que el buffer de recepción serial está lleno o para saber si ya se envió la información.

3. **Revisión del contenido del búffer:** El PIC revisa el contenido del búffer de recepción serial y lo compara con los caracteres especificados para cada estación, es decir, cada estación tiene asociado a él un código que corresponde a la dirección de la estación. Dependiendo del contenido que el PIC encuentra en el búffer, pide un reporte (Monitoreo) o genera una salida que cambia el estado de un relé u otro (Control) de manera de controlar el transmisor.

4. **Respuesta del PIC:** Una vez que el microcontrolador identifica la trama, determina si la solicitud es una petición de un reporte o para control del

transmisor. En caso de que la solicitud sea un reporte el PIC, revisa los puertos de entradas y las envía por su pin de transmisión; en caso de que la petición sea para controlar el transmisor, el PIC activa algún pin de los puertos de salida, y a través de la tarjeta conversora se apaga o enciende el transmisor , enviando luego un reporte para verificar el cambio de estado del transmisor..

Por su parte, las características configuradas en el PC para la comunicación serial fueron las siguientes: una trama de 5 bytes y una velocidad de transmisión de 300 Baudios/seg.

La figura 15 muestra el esquema de conexión del microntrolador PIC16F877.

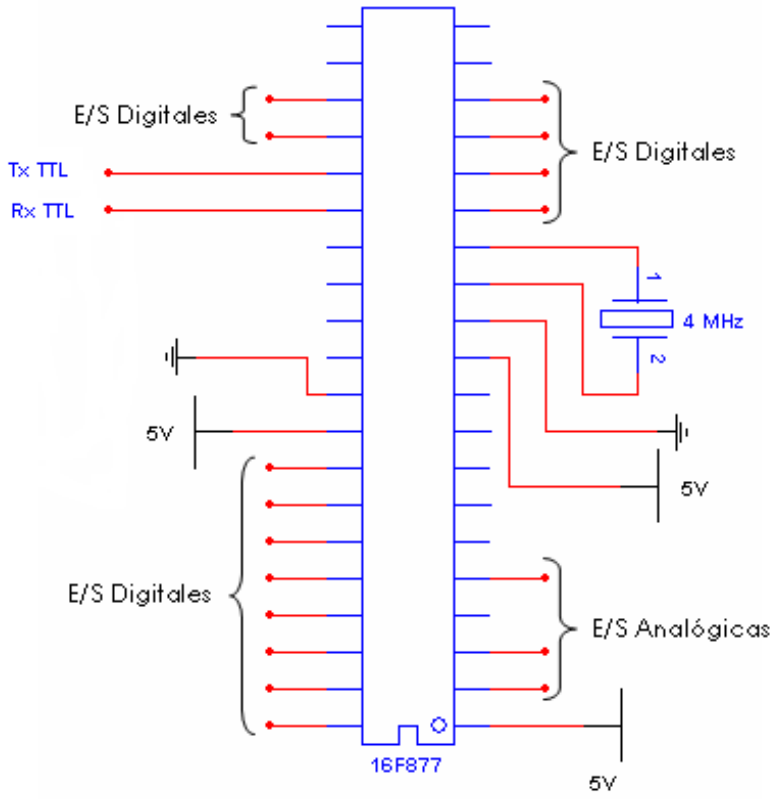


Figura 15. Esquema de Conexión del PIC16F877.

III.3.2 Circuito Conversor

En la Estación Repetidora Piloto (Caricuao), el equipo de ingenieros de Televen procedió a extraer de cada uno de los parámetros a medir la información necesaria, de manera tal, de facilitar el trabajo de conversión y adaptación a la Tarjeta de Adquisición y Control de Datos. Esta información de cada uno de los parámetros fue colocada en un conector DB-25 Hembra de la siguiente manera:

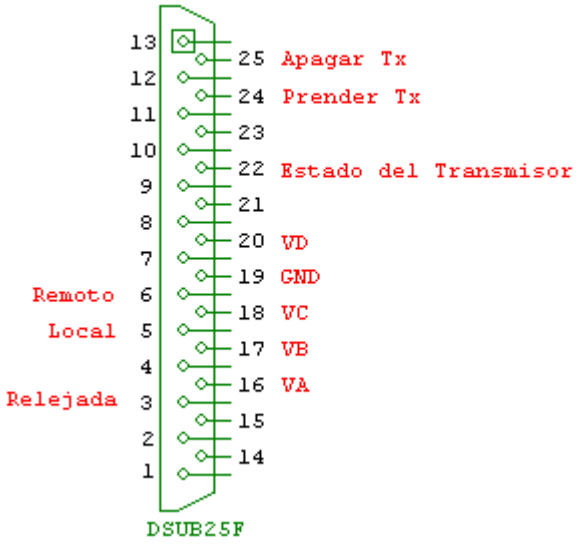


Figura 16. Conector DB-25

Los parámetros restantes, como lo son: Señal de Audio, Señal de Video, Potencia Aural y Potencia Visual, van directamente conectados a la Tarjeta de Adquisición y Control de Datos a través de conectores BNC. El DB-25 mostrado anteriormente, va conectado a la Tarjeta de Adquisición y Control de Datos, para el debido procesamiento de la información.

Para describir los circuitos de conversión que se le aplican a cada uno de los parámetros a medir en el Sistema de Telemetría es necesario realizar una división de los mismos en tres tipos:

III.3.2.1 Entradas Digitales

Los parámetros tomados en cuenta para su adaptación a las entradas digitales del PIC son los siguientes: VA, VB, VC, fuentes de 28V DC (Amplificadores); VD fuente de 28V DC (Driver); Estado del Transmisor; Presencia de Reflejada; Presencia de Audio; Presencia de Video.

El PIC en sus entradas digitales solo permite dos estados, 5V (equivalente a 1 lógico) y Tierra 0V (equivalente a 0 lógico); de tal manera que para mantener la compatibilidad entre el PIC y los parámetros es necesario su adaptación, realizando las conversiones pertinentes en cada uno de ellos.

- **Fuentes DC**

De las fuentes VA, VB, VC, VD, es necesario saber su estado (On/Off). Estos 4 parámetros tienen a su salida un voltaje de 28V en el caso en que estén encendidas, y 0V si están apagadas. De manera de que pueda existir compatibilidad con el PIC, se adaptó los 28V a 5V, utilizando un Relé de 28VDC como se muestra en la figura 17.

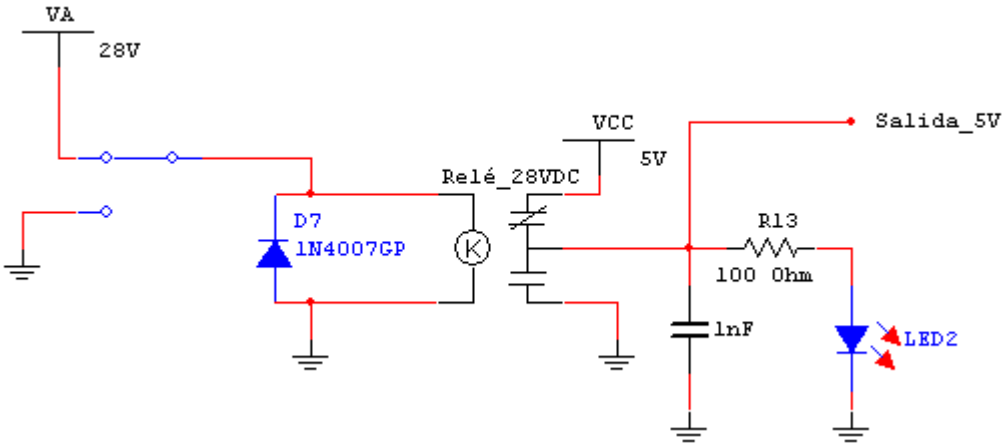


Figura 17. Circuito Adaptador 28V-5V

Para el caso de la presencia de Reflejada se aplica el mismo circuito de la figura 16.

- **Detector de Video**

El circuito de la figura 18 permite accionar un relé al detectar la presencia de una señal de vídeo compuesto en su entrada.

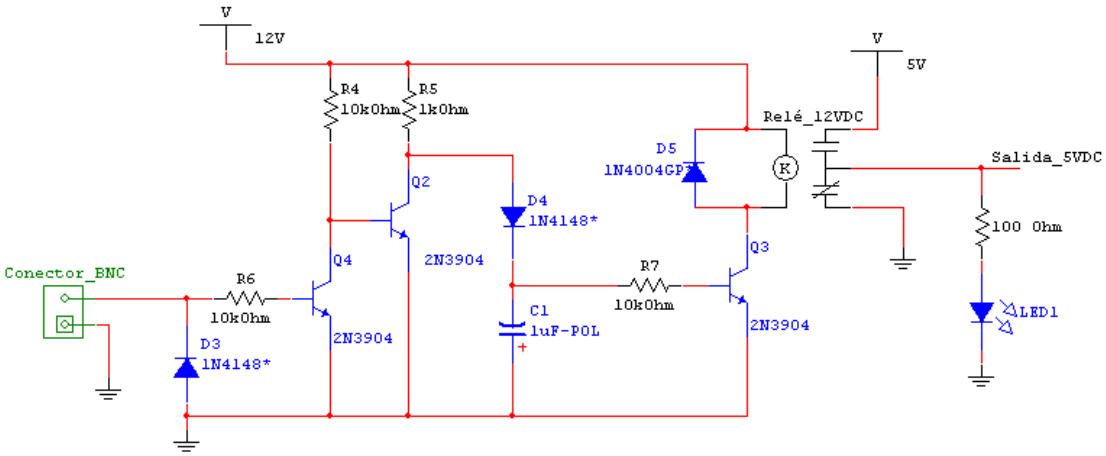


Figura 18 . Detector de Video

El principio de funcionamiento es simple. El diodo en paralelo con la señal de entrada impide el paso de los componentes negativos de la señal de vídeo compuesto. De no limitarse la entrada los pulsos de borrado harían apagar la bobina del relé. El primer transistor hace las veces de preamplificador del segundo, el cual se encarga de cargar el capacitor de 1µF quien hace que el circuito permanezca encendido durante los instantes de borrado. El tercer transistor funciona en corte/saturación accionando la bobina del relé, el cual arroja 5V en el momento en que exista presencia de video.

- **Detector de Audio**

El circuito de la figura 19 permite accionar un relé al detectar la presencia de una señal de audio en su entrada, y lo desactiva cuando pasan algunos segundos de silencio.

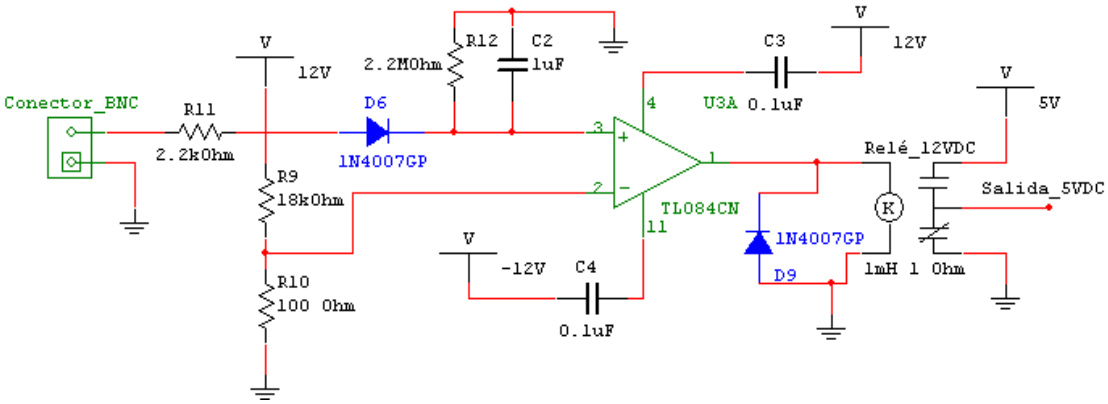


Figura 19. Detector de Audio

El funcionamiento de este circuito se basa en un comparador, el cual recibe en la entrada (-) un voltaje fijado por el divisor resistivo entre las resistencias R9 y R10, y en la entrada (+) se encuentra la señal de Audio luego de ser pasada por un diodo; al estar la entrada (+) a un mayor potencial que el fijado en la entrada (-), la salida procede a activar el Relé, el cual arroja 5V al momento en que exista presencia de Audio.

III.3.2.2 Entradas Analógicas

Las siguientes entradas analógicas serán descritas a continuación: Temperatura de la Sala, Potencia Aural, Potencia Visual.

- **Temperatura de la Sala**

El sensor de temperatura utilizado para esta medición es el LM335, el cual es sumamente sencillo de calibrar para su montaje. Para la calibración es necesario adaptarlo a través del potenciómetro a un voltaje de salida de 2,982V, equivalente a 25ªC; la cual es la temperatura ambiente. Luego de su calibración queda listo para conectar su salida a una de las entradas analógicas del PIC.

Su esquema circuital se muestra en la figura 20:

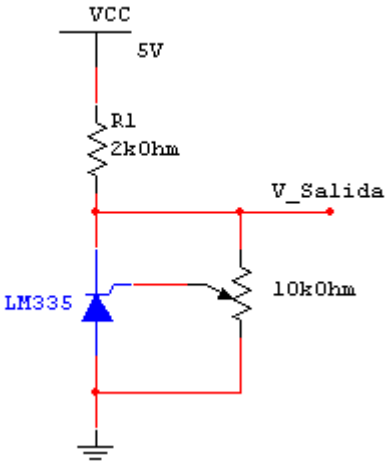


Figura 20. Sensor de temperatura LM335

- **Potencia Aural y Visual**

Para poder medir el valor de la Potencia Aural y la Potencia Visual es necesario filtrar la señal que proviene del transmisor a través de unos filtros externos a la Tarjeta de Adquisición y Control de Datos, véase figura 21, éstos a su vez van conectados a la tarjeta por medio de unos conectores BNC.

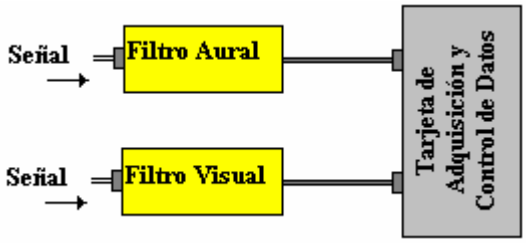


Figura 21. Esquema externo de los filtros para Potencia Aural y Visual.

Al llegar la señal filtrada a la Tarjeta de Adquisición y Control de Datos, se procede a rectificarla para luego ser procesada por el PIC. El circuito rectificador es el siguiente:

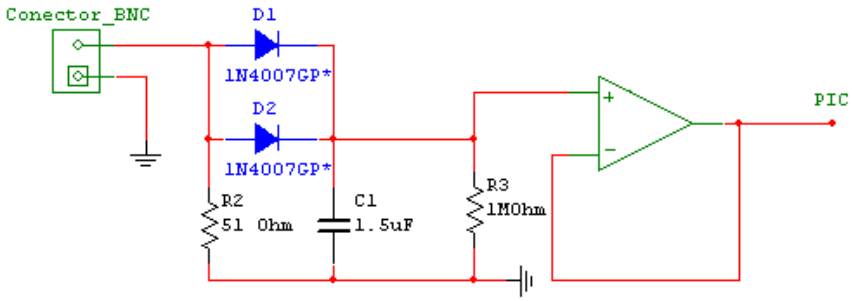


Figura 22. Rectificador para la Potencia Aural y Visual.

Debido a la ausencia de los filtros necesarios para obtener las componentes de la Potencia Aural y Visual, se dejó el diseño de la tarjeta y del software de Monitoreo y Control, de manera tal, que cuando se obtengan los filtros solo sea necesario conectarlos a la Tarjeta de Adquisición y Control de Datos.

III.3.2.3 Salidas

Todas las salidas del PIC, como lo son: Prender Transmisor, apagar Transmisor, Modo Local y Modo Remoto, se controlan de la misma manera. Para cambiar alguno de estos estados, es decir, para apagar/prender el transmisor o para cambiar de Modo Remoto a Local, solo es necesario enviar un bajo (puesta a tierra) a través del canal correspondiente al parámetro. En la figura 23 se observa el esquema realizado para las salidas del PIC.

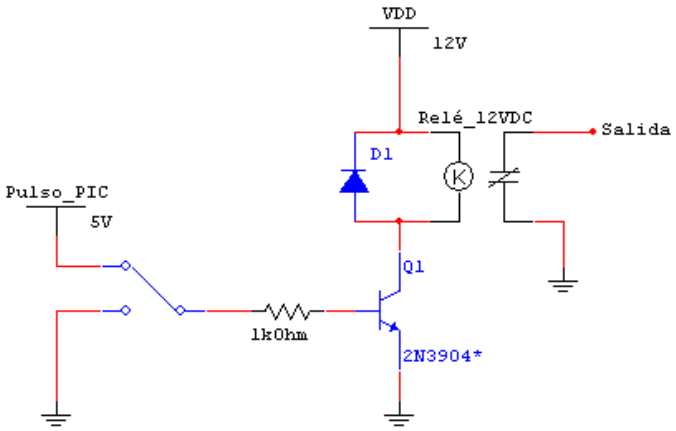


Figura 23. Parámetros de salida

Este proceso consiste en enviar un pulso de 5V desde la salida del PIC pertinente al parámetro del cual se desea cambiar su estado. Este pulso hace saturar al transistor que tiene a su salida, de manera tal, que el Relé se active y permita a la salida una puesta a tierra para lograr el cambio de estado.

III.3.3 Diseño de la Tarjeta de Adquisición y Control de Datos.

El diseño del plano electrónico de la Tarjeta de Adquisición y Control de Datos se realizó bajo el ambiente de MultiSim 7; el diseño está compuesto por el microcontrolador (PIC) y el circuito conversor. En el Apéndice A se encuentra el plano de dicho diseño.

III.3.4 Pruebas de la Tarjeta de Adquisición y Control de Datos.

El diseño de la Tarjeta de Adquisición y Control de Datos se ensambló en un Protoboard, de manera de realizar diversas pruebas que confirmen su funcionamiento, entre las pruebas que se realizaron tenemos:

- Para las fuentes DC se colocó a la entrada del circuito 28V y se midió a la salida 5 V, comprobando el funcionamiento del circuito.

- Para el detector de Audio y Video se simuló a través del equipo PAL/NTSC SIGNAL GENERATOR TECTRONIX tanto la señal de audio como de video, y se colocaron a la entrada de sus respectivos circuitos, al encender el equipo los circuitos detectaban la señal activando el relé y encendiendo el led, confirmando el funcionamiento del circuito.

- Para el sensor de Temperatura bastó con medir el voltaje de salida del sensor, y calentarlo de manera de medir el voltaje y observar variación.

- Para las salidas del PIC, se procedió a activar dichos pines de salida por 1 seg., de manera de observar si los relé que están conectados a dichos pines se activaban, produciendo a la salida la puesta a Tierra necesaria para encender o apagar el transmisor, y para el PTT.

Es importante resaltar que dichas pruebas también se realizaron con los circuitos impresos de las Tarjetas en ambos casos las pruebas fueron exitosas.

III.4 Sistema de Transporte de Datos

El principal obstáculo planteado en el proyecto fue el de cómo transportar la data desde y hasta la central utilizando el sistema de radio instalado actualmente, ya que constituía para la Corporación Televen el proyecto más económico debido a que no se requeriría de una inversión cuantiosa de dinero en equipos nuevos.

A pesar de lo propuesto inicialmente por Televen se consideraron algunas alternativas a tal propuesta, tales como: Modems telefónicos, Radio Modems, mensajes de texto y los Radios UHF.

Después de alcanzar los resultados del estudio realizado con las diferentes propuestas, se llegó a la conclusión de utilizar para la implementación del Sistema de Transporte los Radios UHF.

El Sistema de Transporte de Datos está conformado por los radios Motorola UHF y la Tarjeta de Transporte. En general, el Sistema se encarga de la modulación de la data enviada por el Sistema de Monitoreo y Control (PC Central), para luego ser transmitida vía radio UHF, de manera que éste envíe la data a través de la red de antenas de Televen. Al llegar la data a la Estación Repetidora de destino, se procede a la demodulación de la misma, para que sea procesada y analizada por el PIC. Una

vez procesada, el PIC procede a enviar una respuesta, con la información solicitada de los parámetros. Ocurriendo el mismo proceso de modulación y demodulación desde el PIC hasta la Central.

III.4.1 Radios UHF

Los radios UHF Motorota mostrados en la figura 24 son los modelos más comunes presentes en las estaciones repetidoras y en la Central de Operaciones de Televen. La desventaja que presentan estos radios, es que no soportan el transporte de data digital, sino de señales analógicas con componentes frecuenciales hasta 4KHz.

(a) *(b)*

Figura 24. Equipos de radio Motorola Radius: *(a)GM300, (b) M120*

Para poder llegar a esta conclusión, se realizaron pruebas preliminares con los radios, las cuales se describen brevemente a continuación:

a) Envío de la data TTL utilizando la modulación del radio

Para los fines de esta prueba se implementó un circuito que convierte la señal que sale del puerto serial de la PC (cuyos valores de voltaje están entre -

10 y 10 voltios) en una señal TTL (cuyos valores están entre 0 y 5 voltios) mediante el uso del integrado MAX232. La señal que sale del integrado se colocó a la entrada del radio transmisor (Motorota Radius M120) y para la recepción se tomó la señal que sale del radio receptor (Motorota Radius GM300) y se implemento un circuito de amplificación bajo saturación (usando el integrado LM741), lo que producía una señal perfectamente cuadrada y sin DC (el nivel DC se lo quitaba alguno de los dos radios) cuyos valores estaban entre -10 y 10 voltios después de la etapa de amplificación. Cabe destacar que estos niveles de voltaje son muy importantes para que el puerto serial de la PC trabaje correctamente. La señal que sale del circuito de amplificación se colocó a la entrada del puerto serial de la PC y mediante el uso del software Mscomm se detecta la entrada del puerto de comunicaciones.

Se probó con el envío de varias cadenas de caracteres y la única que llegaba de manera correcta era la que estaba compuesta por una serie continua de letras U porque la representación ASCII de esta letra es 10101010, por lo que el puerto en cualquier momento se sincronizaba. Esta prueba permitió concluir que los radios utilizados no funcionaban correctamente para tal señal digital.

b) Envío de la data usando un hardware de modulación digital

Dados los inconvenientes que presentaron los radios con el transporte de data digital, se procedió a diseñar un circuito modulador que permitiera adaptar la señal binaria EIA-RS232 a los mismos para que pudieran transmitirla como una señal analógica. Se optó por la implementación de un modulador FSK, debido a su gran ventaja que presenta para la inmunidad al ruido y a las interferencias, fundamental para transmitir data vía radio. Después de las pruebas realizadas con los equipos de radio, se decidió realizar el diseño completo del hardware capaz de modular la señal digital que sale del puerto de comunicaciones de la PC y adaptar dicha señal a los equipos de radio.

Por otra parte, el radio Motorola posee un conector de 16 pines, de los cuales se utilizaron 4 pines: Transmisión (Tx), Recepción (Rx), Tierra (GND) y Push to talk (PTT). Los radios UHF poseen la capacidad de utilizar 5 canales, para el enlace de comunicaciones se utilizó el canal 1.

Con la ayuda del software Radio Mobile, se diseñó el enlace principal que conecta a la Central de Televen con la Estación Repetidora de Caricuao, la cual, es la Estación Piloto en donde se realizaron todas las pruebas pertinentes a este proyecto; En la figura 25 se visualiza el enlace mencionado.

Figura 25. Enlace Televen – El Cuño – Caricuao.

Es importante recalcar que la Central de Televen utiliza la Repetidora principal del Cuño para salir a cualquier otra repetidora, esto es debido a su ubicación estratégica en el Cerro el Ávila para lograr una gran línea de vista a todo el Valle de Caracas y a otros puntos de repetición. A continuación se muestra la línea de vista que tiene Televen-El Cuño-Caricuao.

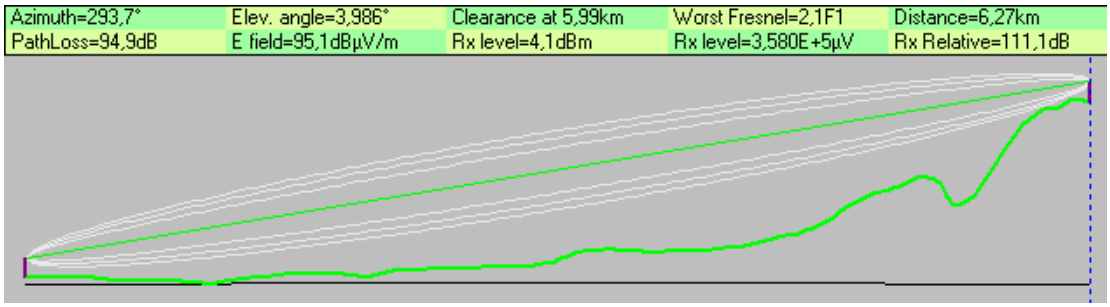


Figura 26. Línea de Vista Televen – EL Cuño.

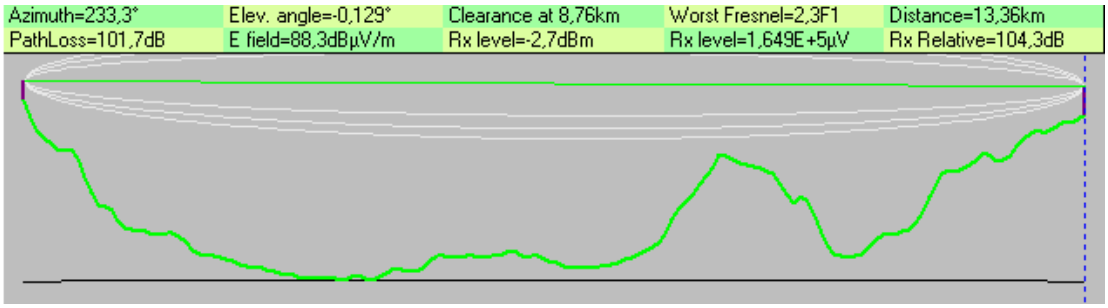


Figura 27. Línea de Vista El Cuño- Caricuao.

III.4.2 Tarjeta de Transporte

La tarjeta de transporte consta de un modulador y demodulador FSK, dicha tarjeta se encuentra tanto en la Estación Central como en la Estación Repetidora, con la diferencia que la tarjeta ubicada en la estación Central posee un circuito conversor RS-232 a TTL, necesario para adaptar las señales entre la PC y el módem.

III.4.2.1 Modulador FSK

Para la modulación FSK se utilizó el chip XR-2206 ya que puede ser utilizado como un generador de FSK, para esto es necesario colocar 2 resistencias R1 y R2 en los pines 7 y 8, respectivamente, como se muestra en la figura 28. Dependiendo de la polaridad lógica de la señal de entrada (Pin 9) uno de los pines 7 y 8 será activado. Si la señal de entrada es un circuito abierto o el nivel de voltaje esta por encima de los

2V se activará la resistencia R1. Similarmente si el nivel de voltaje del pin 9 es menor o igual a 1V se activará R2. Para calcular R1 y R2 se utilizan las siguientes fórmulas.

$$f_1 = \frac{1}{R_1.C} \qquad (3.1)$$

$$f_2 = \frac{1}{R_2.C} \qquad (3.2)$$

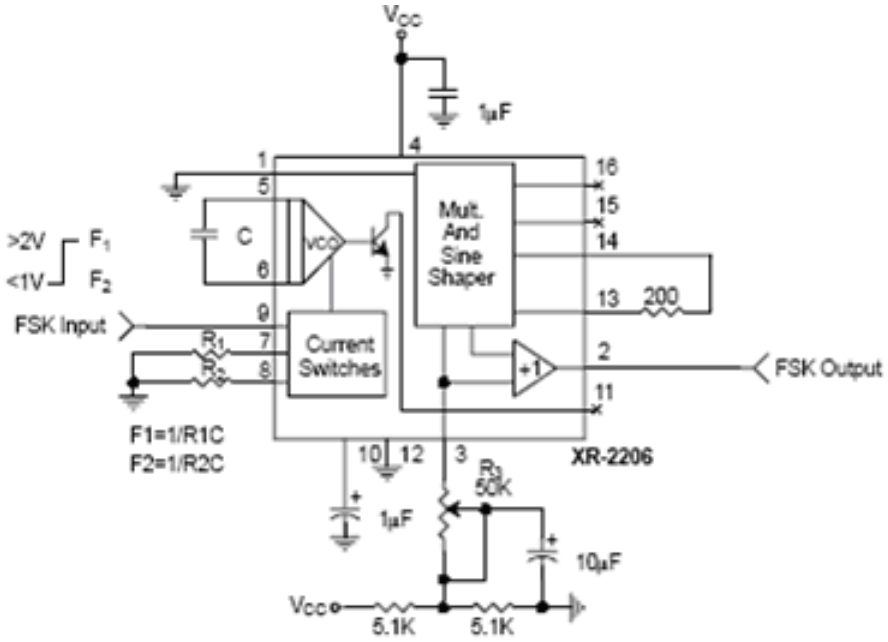


Figura 28. Montaje Modulador FSK.

Las frecuencias escogidas para *f1* y *f2* son 1200Hz y 2200Hz respectivamente, fijando el capacitor a C= 1nF, y aplicando la fórmula 3.1 y 3.2:

R1= 470K; y R2= 840K.

III.4.2.2 Demodulador FSK

Para la demodulación se utilizó el chip XR2211, para esto fue necesario realizar el montaje de la figura 29, pero antes era necesario realizar ciertos cálculos de los

componentes que se utilizarían en dicho montaje, estos cálculos nos permiten fijar la frecuencia central con la cual el chip va a comparar la señal de entrada, de manera que para frecuencias mayores la salida sea un 1 lógico y para frecuencias menores sea un 0 lógico. A continuación se presentan dichas fórmulas:

$$f_o = \sqrt{f_1 . f_2} \qquad (3.3)$$

$$R_1 = \frac{R_o . f_o}{(f_1 - f_2)} .2 \qquad (3.4)$$

$$R_o = R_o + \frac{R_x}{2} \qquad (3.5)$$

$$C_f = \frac{0,25}{R_{sum} . BaudRate} \qquad (3.6)$$

$$R_f = 5 . R_1 \qquad (3.7)$$

$$R_b = 5 . R_f \qquad (3.8)$$

$$C_1 = \frac{1250 . C_0}{R_1 . 0,25} \qquad (3.9)$$

Aplicando cada una de estas fórmulas los valores a utilizar serían los siguientes:

Fo = 1624; *R1* = 51K; *Cf* = 1nF.

Rf = 255K; *Rb* = 1.2M; *C1* = 3.9nF.

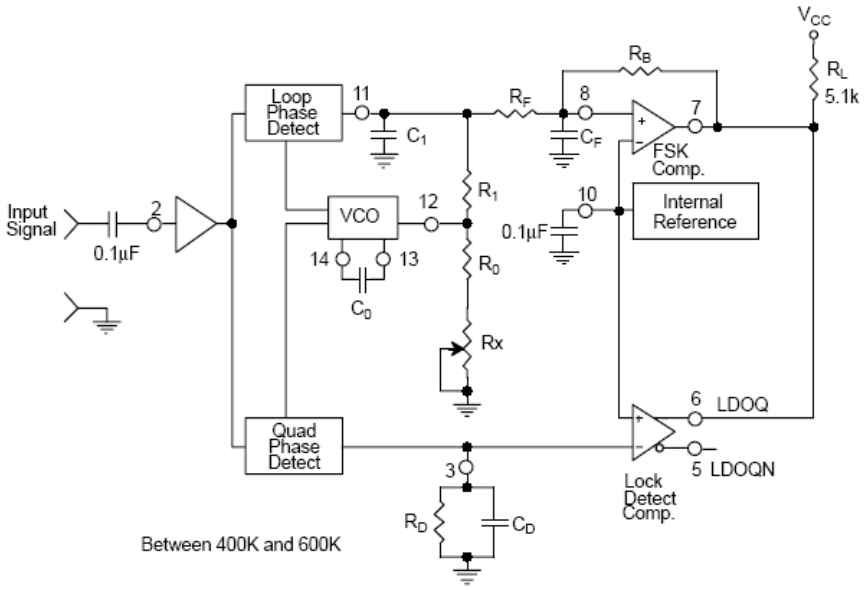


Figura 29. Montaje Demodulador FSK.

III.4.2.3 Circuito Conversor RS232-TTL.

El circuito conversor RS232-TTL permite adaptar las interfaces de la PC (RS232) con el módem FSK (TTL), para este circuito conversor se utilizó el chip MAX-232, el cual a través del montaje de la figura 30 cumple con dicha función.

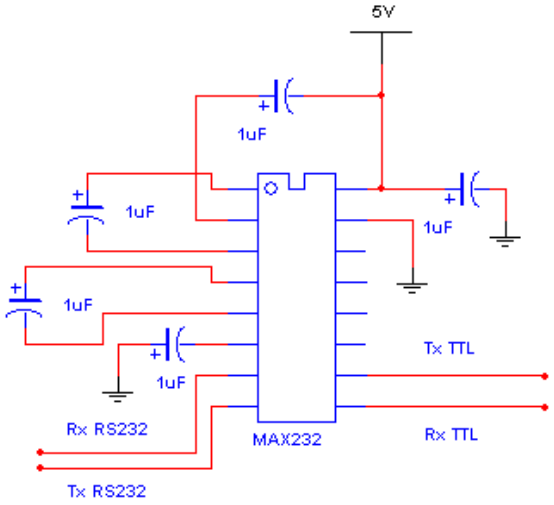


Figura 30. Circuito Conversor RS232-TTL

III.4.2.4 Diseño de Tarjeta de la Estación Central y Repetidora.

En el Apéndice B y C se pueden observar los planos electrónicos de dichas tarjetas, las cuales están compuestas por un modulador FSK (XR2206), un demodulador FSK (XR2211), un circuito de PTT, un circuito conversor RS232-TTL (MAX232), explicados anteriormente. El circuito conversor RS232-TTL, solo se encuentra en la Tarjeta de la Estación Central debido a que en ésta se localiza la PC.

III.4.2.5 Pruebas Tarjetas de la Estación Central y Repetidora.

Al realizar el diseño de las tarjetas se procedió a efectuar distintas pruebas que confirmen su funcionamiento, para realizar estas pruebas el diseño se ensambló en un Protoboard. A continuación se explicarán dichas pruebas.

- Para el conversor RS-232-TTL bastó con enviar una señal por el puerto RS232 del PC (±12), y medir con un osciloscopio a la salida del circuito conversor la señal TTL.

- Para el modulador XR2206 se procedió a colocar una señal TTL a la entrada y observar a través del osciloscopio la salida modulada.

- Para el demodulador XR2211 se procedió a puentear la salida del modulador a la entrada del demodulador, y observar a través del osciloscopio la señal demodulada.

- Para el circuito del PTT, se colocó la entrada del circuito al pin DTR del puerto, de manera que al abrir el puerto se activara el relé y al cerrarlo se desactivara.

- Se procedió en enviar por el puerto RS232 una cadena de caracteres, de manera de que esta fuera modulada, demodulada y llegara de nuevo al puerto RS232, probando así todos los elementos de la Tarjeta.

- Se configuraron los radios UHF punto a punto, y se envió una cadena de caracteres, la cual se modulaba y se enviaba por el radio Tx, luego el radio receptor recibía la señal, la demodulaba de manera que esta llegara de nuevo a la PC y se mostrara en pantalla la cadena de caracteres enviada.

- La prueba anterior se repitió pero con los radios UHF configurados punto multipunto, en la cual la señal llegaba hasta el Cuño y regresaba.

Es importante resaltar que dichas pruebas también se realizaron con los circuitos impresos, en ambos casos las pruebas fueron exitosas.

Capítulo IV

Resultados

Los resultados obtenidos de las fases de la Metodología para el Diseño e Implementación del Sistema de Telemetría para Controlar las Estaciones Repetidoras de Televen se describen a continuación:

IV.1 Sistema de Monitoreo y Control.

IV.1.1 Mscomm.

A continuación se muestra la información recibida por el computador previo envío de la trama cuando se solicita un reporte.

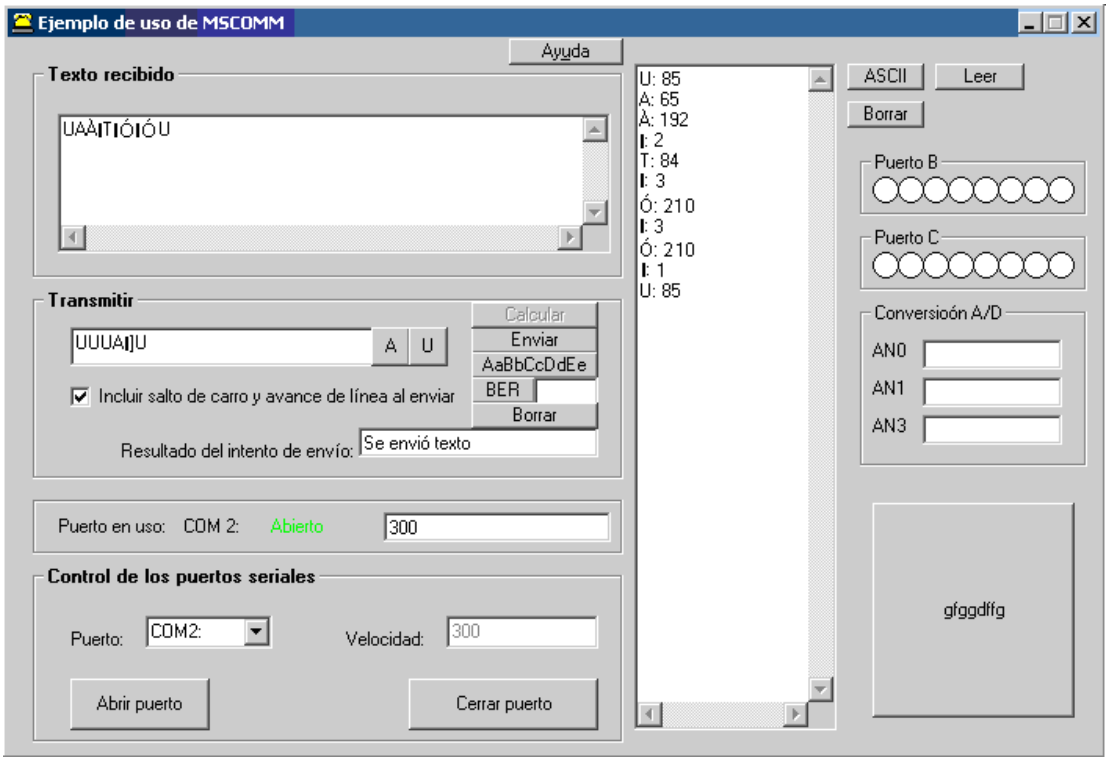


Figura 31. La trama enviada con recepción exitosa.

Como se puede observar en la figura 31 los caracteres transmitidos (campo Transmitir) mediante la interfaz serial RS232 (COM2) corresponden a los descritos en la estructura de la trama, Tabla 3.

Carácter	ASCII	Binario
U	85	01010101
A	65	01000001
|	28	00011100
]	93	01011101
U	1	01010101

Tabla 3. Representación en ASCII de la data enviada.

El significado de cada uno de estos caracteres es el siguiente: una "U" correspondiente al Bit de Inicio, seguido de una "A" el cual identifica a la Estación Repetidora, un "|" que en este caso es el byte correspondiente a la orden "solicitar reporte", un "]" el cual es el Checksum correspondiente a la suma del ID con el byte destinado para la orden de solicitar reporte, y una "U" como Bit de parada. Esta data se envió usando una velocidad de 300bps, que aunque pueda parecer baja para algunas aplicaciones modernas, es ideal para este tipo de aplicaciones debido a la restricción que imponen los radios en cuanto a la frecuencia de corte de sus filtros.

Una vez transmitida la data mediante el puerto serial, se moduló mediante un hardware específico para ser transmitida vía radio y recibida por todas las estaciones. Como respuesta a dicho evento, la Tarjeta de Adquisición y Control de Datos es quien responde al identificar la trama como una petición de reporte desde la Central, lo cual es lo que se puede observar en el campo **Texto recibido**, donde cada uno de ellos tiene un significado; algunos representan conversiones analógicas a digitales

provenientes de algunas mediciones y otros son un grupo de bits que indican el estado de encendido o apagado de algún equipo o algún parámetro específico que no requiere ningún tipo de medición adicional. Para el caso que se muestra en la figura 31 los resultados son los siguientes:

Carácter	ASCII	Binario
U	85	01010101
ÿ	255	11111111
À	192	1100000
|	2	00000010
T	84	01010100
|	3	00000011
Ò	210	11010010
|	3	00000011
Ò	210	11010010
|	1	00000001
U	85	01010101

Tabla 4. Representación en ASCII de la data recibida.

En la Tabla 4 se pueden apreciar que se recibe una cadena de 11 bytes (88 bits), de la cual los bytes (letras “ÿ” y “À” para este caso) representan bit a bit el estado de encendido o apagado de algún equipo o parámetro especifico. Un uno binario representa encendido o enganchado y un cero apagado o desenganchado, como por ejemplo el estado de una fuente de alimentación.

De los 16 bits correspondientes a la lectura de los puertos B y C del PIC, se disponen de 14 bits útiles ya que 2 de ellos representan el estado de la comunicación

vía puerto serial, debido a que dos de los pines del puerto C del PIC está relacionado con su USART, por lo que el valor de los bits más significativos del segundo octeto recibido (11xxxxxx) están relacionados con el estado de la comunicación PIC-PC y no se toman en cuenta como una medición realizada.

El resto de los octetos son utilizados en pares para representar las lecturas analógicas obtenidas por el PIC y convertidos en digitales (Tarjeta de Adquisición y Control de Datos). Debido a que el PIC trabaja con registros de 8 bits, pero su resolución de conversión de 10 bits (con un error de medición de apenas 4,6 mV) es necesario utilizar dos registros para almacenar la conversión: Del primer byte sólo son necesarios los dos menos significativos que representan los dos bits más significativos de la conversión junto al segundo byte. Ejemplo: xxxxxx0100011010 equivale a 1.37V, para realizar la conversión es necesario conocer la resolución (1024) y el voltaje de referencia, que en este caso es de 5V (alimentación).

El mismo procedimiento se aplica para los cuatro bytes restantes, lo cual indica que hay un total de 3 conversiones analógico/digital, relacionadas a la y Temperatura de la sala, Potencia Visual, Y Potencia Aural.

IV.1.2 Interfaz Final

Posterior a la realización de todas las pruebas requeridas se procedió a adaptar MsComm a una aplicación que interprete los resultados obtenidos como respuesta de las peticiones enviadas por el operador, buscando que el diseño final tuviera una interfaz amigable y representara de una manera coherente y confiable los datos obtenidos por la Tarjeta de Adquisición y Control de Datos.

Como se puede observar en la figura 32 se presenta el mapa de Venezuela con la ubicación de todos los estados que poseen estaciones repetidoras identificándolos con el logotipo de Televen.

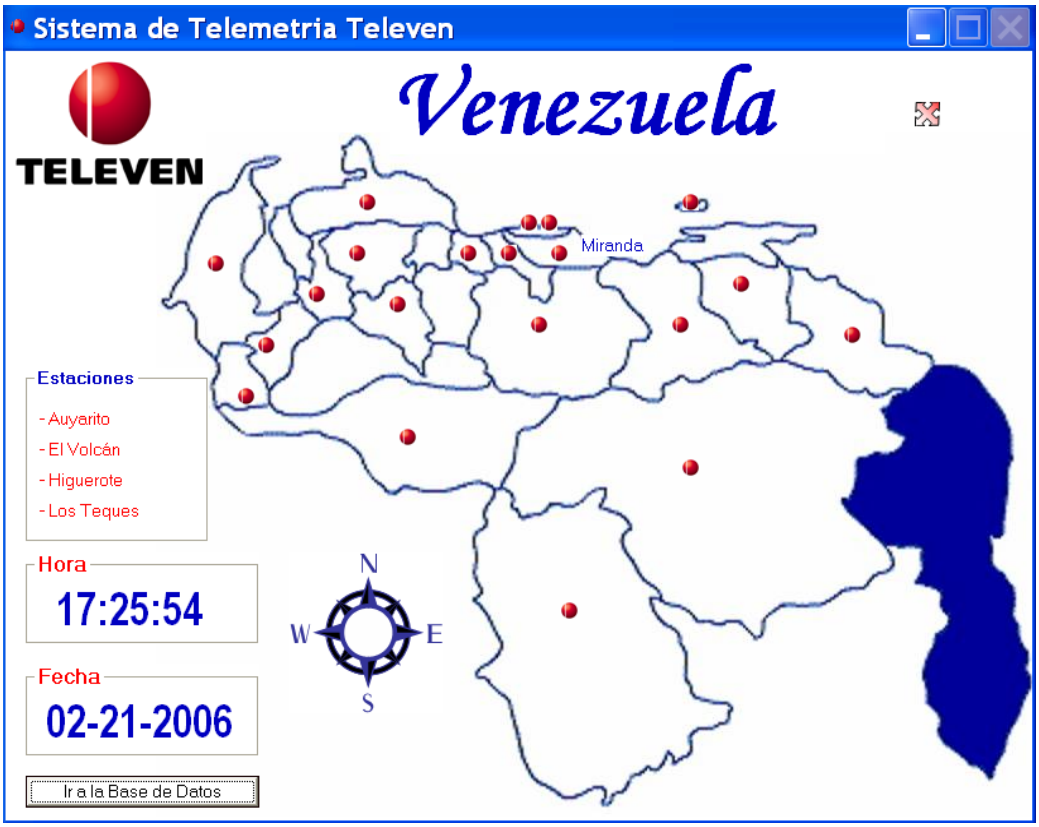


Figura 32. Visión principal del programa de Monitoreo y Control

Se puede visualizar el nombre de las estaciones de cada estado con solo fijar el cursor del mouse sobre el logotipo, además de ofrecer la hora y la fecha del día actual.

Asimismo la interfaz tiene un botón que permite ir a la base de datos, en donde se tienen archivados los reportes solicitados a cada estación y sus respectivas gráficas de algunos de los parámetros más importantes para la observación de su comportamiento. Esta interfase se abre al oprimir el botón que dice “Ir a la Base de Datos”.

Al hacer clic en alguno de los logos de Televen ubicados en los Estados aparecerán interfases como las siguientes:

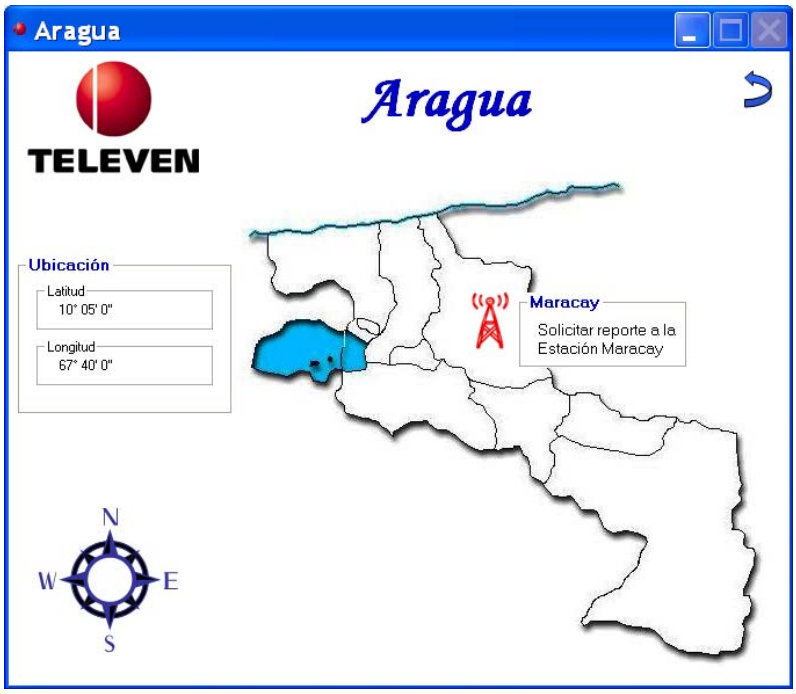


Figura 33. Interfaz Estado Aragua

Figura 34. Interfaz Estado Nueva Esparta

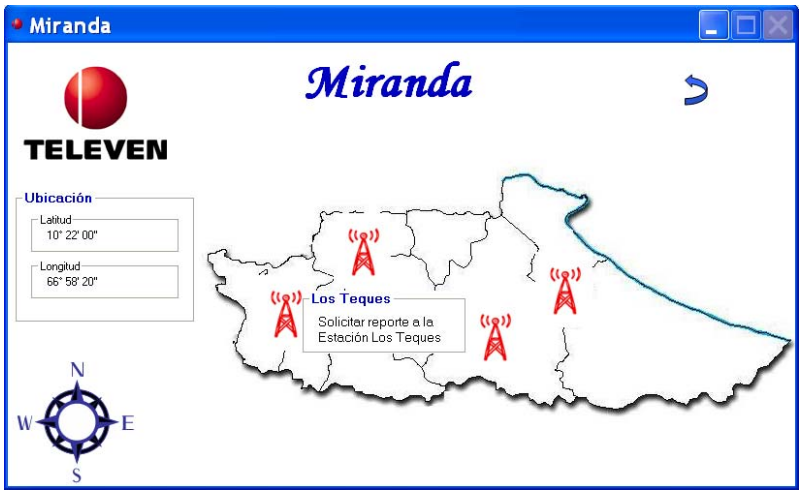


Figura 35. Interfaz Estado Miranda

En cada una de las interfases de los Estados se encuentran ubicadas las repetidoras del mismo, simbolizadas con una antena transmisora. Esta interfaz ofrece la indicación de pedir el reporte de la estación al hacer clic sobre ella.

IV.1.3 Interfaz de Transporte

Al momento de solicitar un reporte a una estación se ilustra en la pantalla la ventana expuesta en la figura 36.

Figura 36. Ventana de Reporte

Este reporte se pidió a la Estación Repetidora de Caricuao (Estación Piloto), el día en que se instaló el sistema.

En la Figura 36 se observan varias divisiones en la ventana entre las cuales se encuentra: Entradas Digitales, Entradas Analógicas, Características y Transmisor.

La primera división que se observa es la de Entradas Digitales, en ésta se encuentran ubicados cada uno de los parámetros de la estación de los cuales se desee saber su estado (Encendido/Apagado, Presencia o no de la señal de audio o video, etc), en donde el circulo verde significa "si" y el rojo "no", es decir, para el parámetro del transmisor, verde significa que está encendido y rojo lo contrario. En la división de entradas analógicas se ubican los parámetros que se deseen medir (Potencia Aural, Visual, Temperatura). Ambas divisiones de la ventana (Entradas Analógicas y Entradas Digitales) pertenecen a la parte de Monitoreo de las estaciones.

Asimismo al observar la división de Características, aquí se coloca el Estado y Estación a la cual se le pidió el reporte y la fecha y hora del mismo, esta división es de gran utilidad a la hora de imprimir el reporte.

Si se observa la parte superior derecha de la ventana se encuentra la división de Control (Transmisor) de las estaciones, en la cual se controla el transmisor de la estación repetidora a la cual se le pide el reporte, esta división es zona restringida, es decir, solo personal autorizado puede habilitarla como se observa en la figura 37. Para poder habilitar las funciones de Control del Transmisor es necesario de un login y una clave, la cual es secreta y única para cada usuario, y esta le permitirá habilitar las funciones que controlan el transmisor como lo son: el encendido, apagado y reset del mismo.

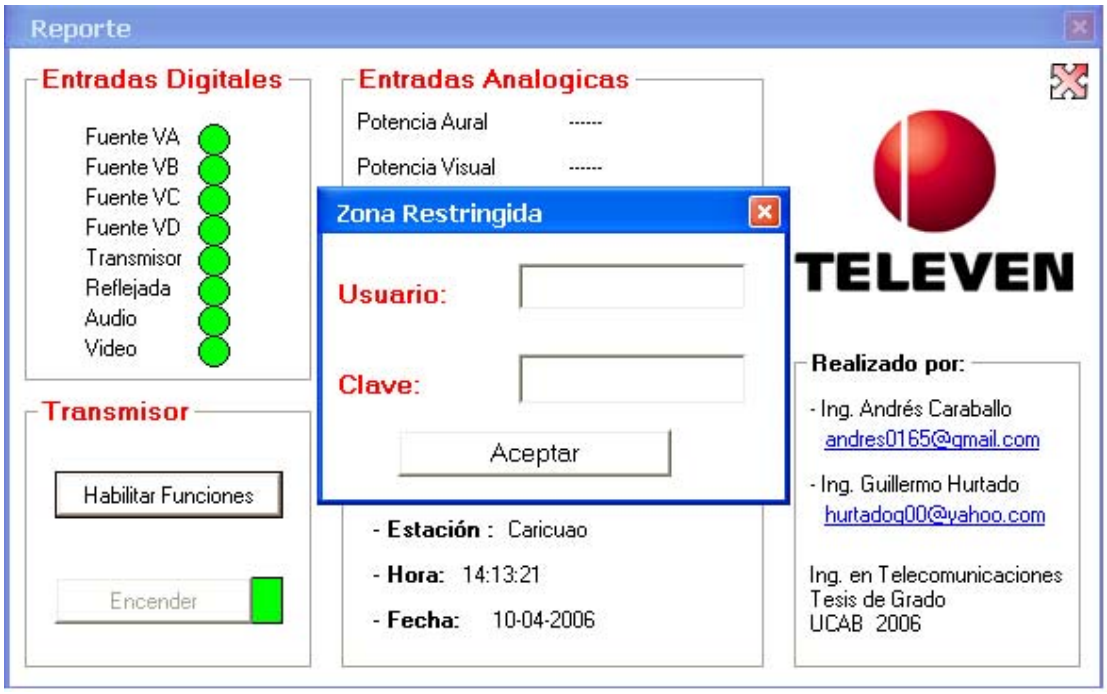


Figura 37. Ventana de Zona Restringida.

Cuando el reporte se solicita aparece una ventana de Información o Advertencia indicando si el reporte fue completado con éxito o si el reporte ha sido fallido, como se expone en la figura 38.

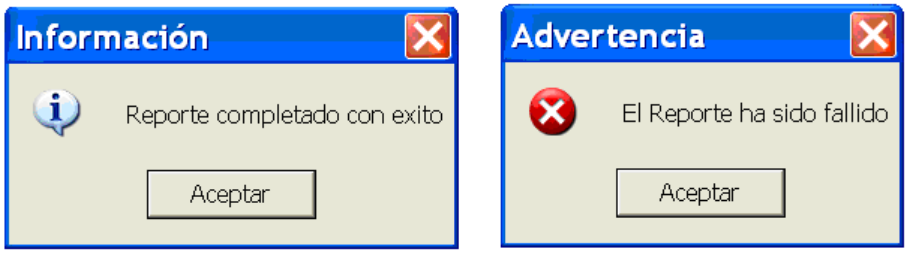


Figura 38. a) Ventana de Información, b) Ventana de Advertencia.

El Sistema de Monitoreo y Control permite guardar e imprimir el reporte que se ha solicitado, para esto el usuario debe hacer clic en el botón de Guardar o Imprimir ubicados en la Ventana del Reporte. En la figura 39 se observa el momento en que se

solicita imprimir el reporte, esta opción imprime la ventana completa con todos los parámetros y características.

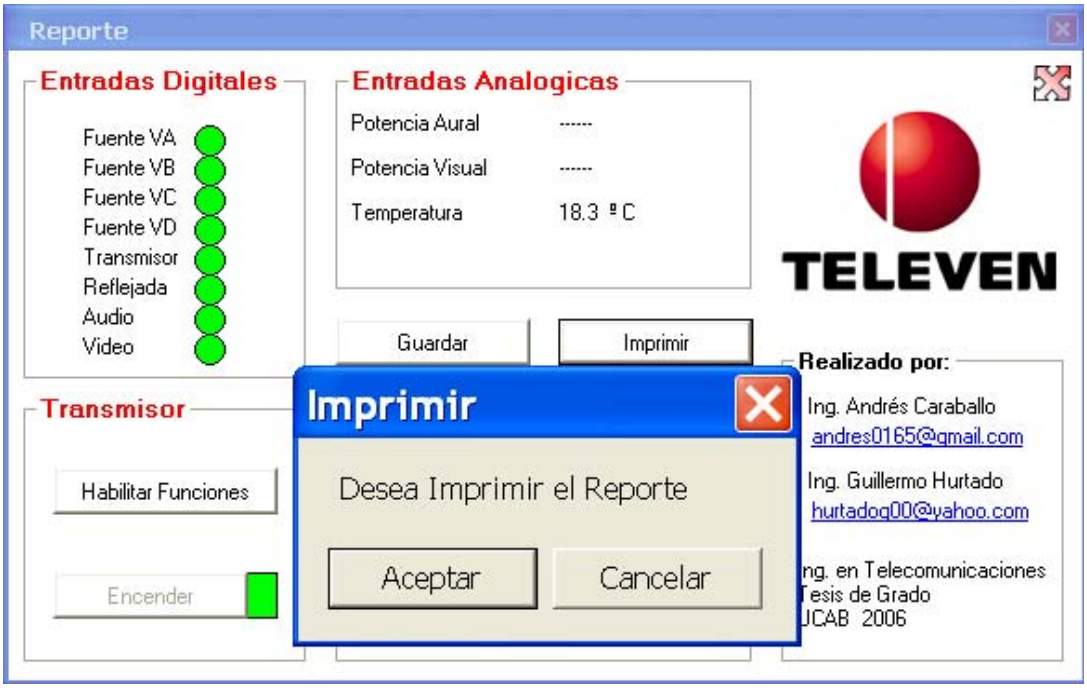


Figura 39. Impresión del Reporte

En cambio al hacer clic sobre el botón Guardar el reporte solicitado automáticamente será guardado en un Libro de Excel, el cual almacenará todos los reportes solicitados que sean guardados como se observa en la figura 40.

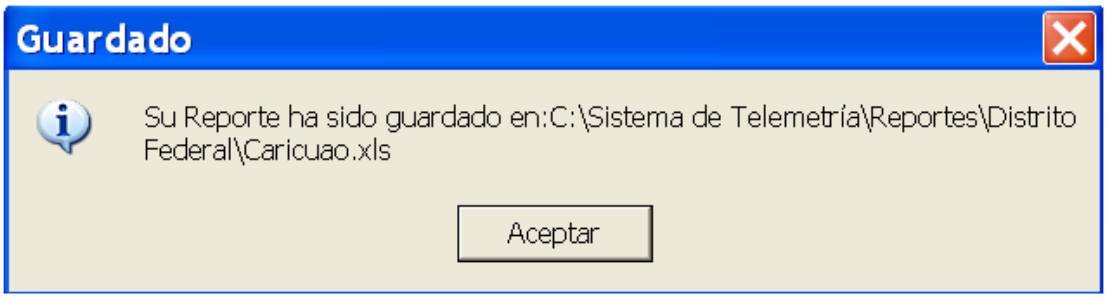


Figura 40. Guardar el Reporte.

IV.1.4 Interfaz Base de Datos.

Para acceder a la Base de Datos es necesario hacer clic en el botón "Ir a la Base de Datos" ubicado en la Ventana Principal del Sistema de Monitoreo y Control, inmediatamente aparecerá la ventana Base de Datos como se muestra en la figura 41.

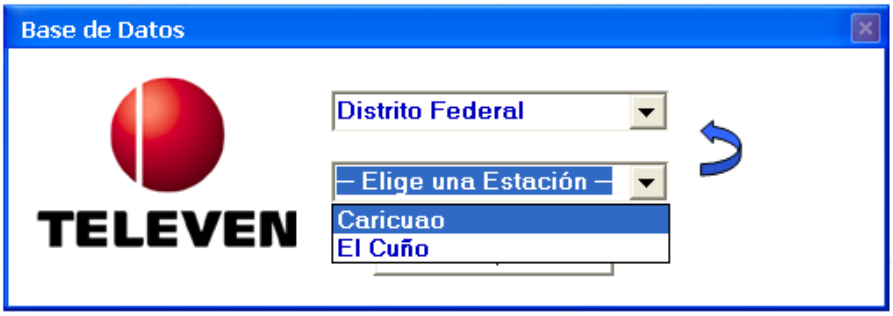
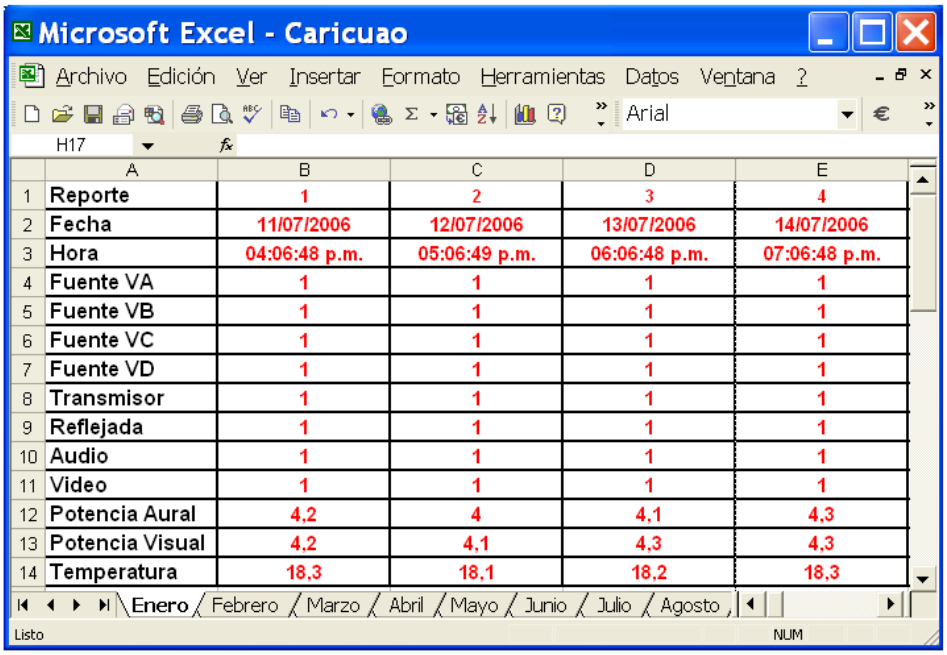


Figura 41. Base de Datos

La ventana de Base de Datos permite elegir el estado y la estación en el cual se quiere buscar el reporte, luego de hacer la selección automáticamente se abrirá la aplicación Excel en el libro deseado, como se muestra en la figura 42.



	A	B	C	D	E
1	Reporte	1	2	3	4
2	Fecha	11/07/2006	12/07/2006	13/07/2006	14/07/2006
3	Hora	04:06:48 p.m.	05:06:49 p.m.	06:06:48 p.m.	07:06:48 p.m.
4	Fuente VA	1	1	1	1
5	Fuente VB	1	1	1	1
6	Fuente VC	1	1	1	1
7	Fuente VD	1	1	1	1
8	Transmisor	1	1	1	1
9	Reflejada	1	1	1	1
10	Audio	1	1	1	1
11	Video	1	1	1	1
12	Potencia Aural	4,2	4	4,1	4,3
13	Potencia Visual	4,2	4,1	4,3	4,3
14	Temperatura	18,3	18,1	18,2	18,3

Figura 42. Base de Datos Hoja Excel

Al mismo tiempo la Base de Datos ofrece al usuario las gráficas dinámicas pertinentes a los parámetros analógicos, ver figura 43.

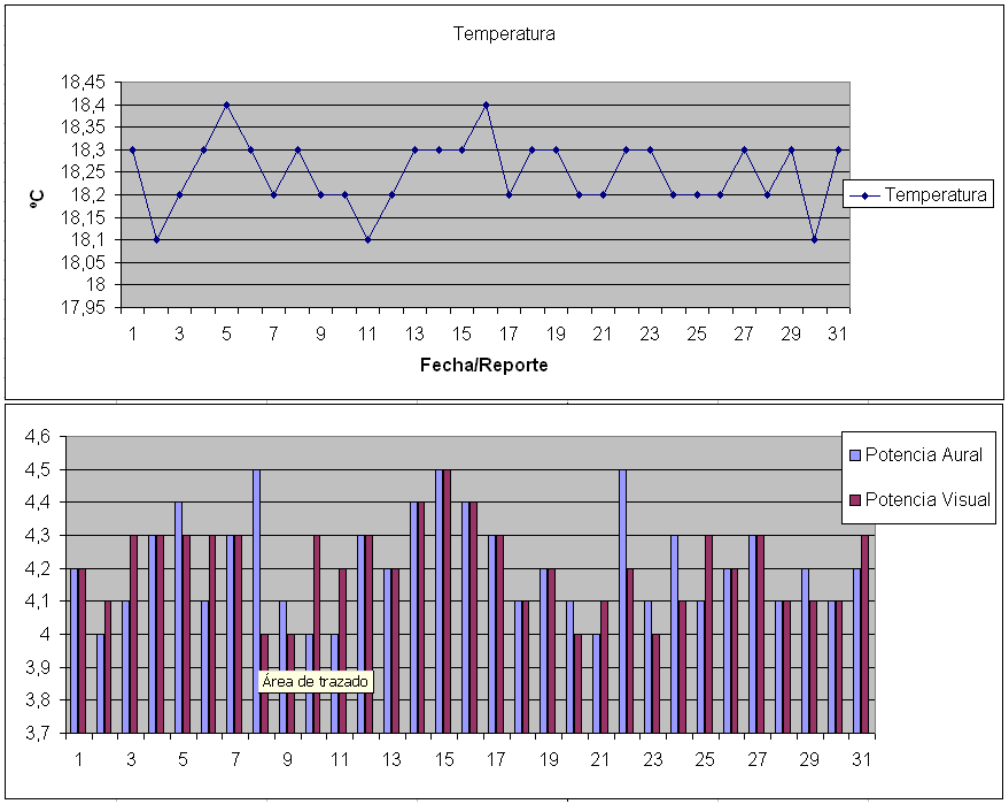


Figura 43. Gráficas de la Base de Datos

IV.2 Tarjeta de Adquisición y Control de Datos.

Luego de realizar el plano eléctrico de las tarjetas se procedió a elaborar los circuitos impresos, para la elaboración de los mismos Televen recomendó que dichos circuitos impresos fueran elaborados por una Compañía especializada en circuitos impresos, ya que en Televen no contaban con los materiales necesarios y apropiados para realizarlas, de manera que la parte de la elaboración de los circuitos impresos y la caja metálica en donde van ubicados fue realizado por la compañía Electro Mecánica Collmer. A continuación se presenta en las figuras 44 y 45 los circuitos impresos finales y la distribución de los materiales de la Tarjeta de Adquisición y Control de Datos.

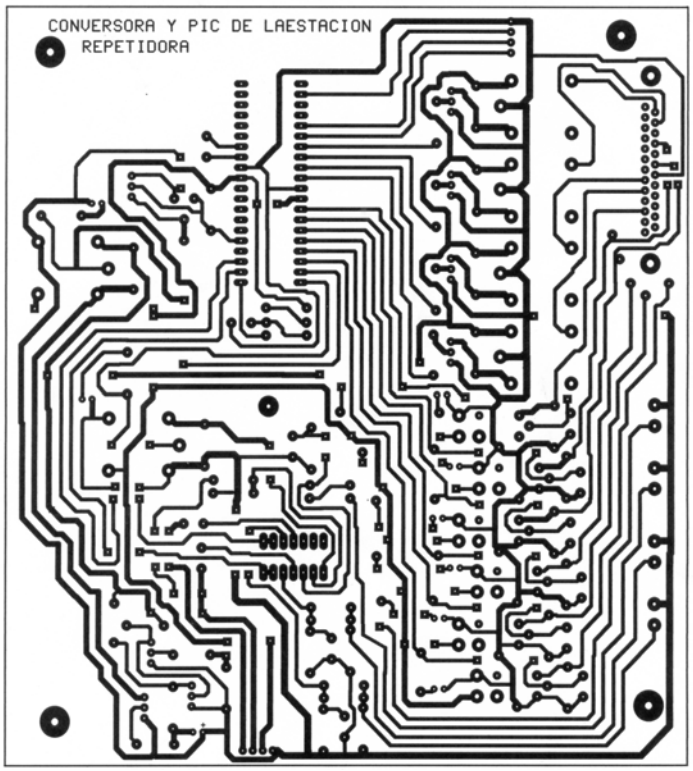


Figura 44. Circuito Impreso de la Tarjeta de Adquisición y Control de Datos.

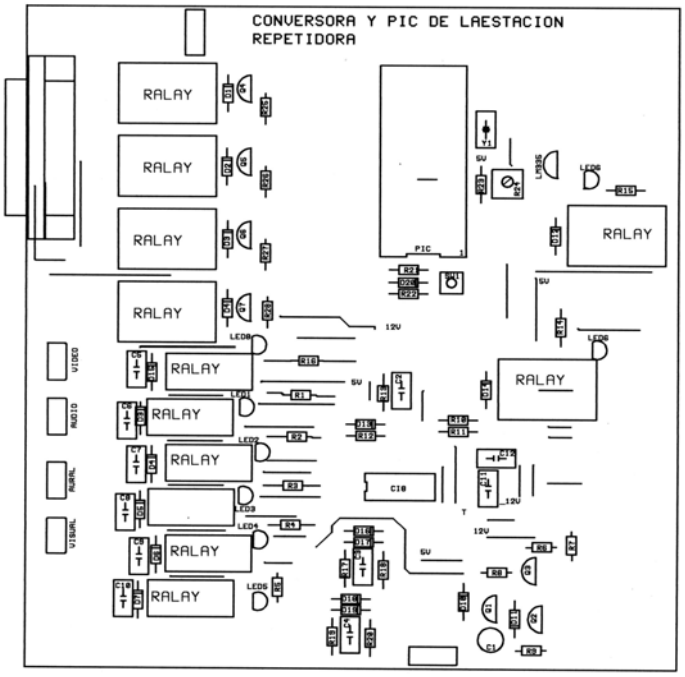


Figura 45. Distribución de la Tarjeta de Adquisición y Control de Datos.

De la misma manera se presenta en la figura 46, el acabado final de la Tarjeta de Adquisición y Control de Datos.

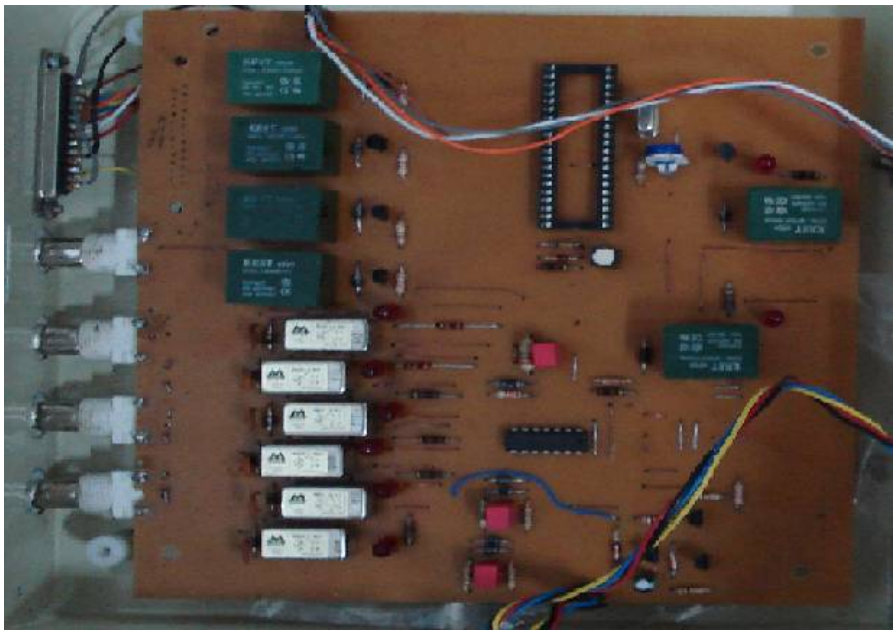


Figura 46. Acabado final de la Tarjeta de Adquisición y Control de Datos

IV.3 Sistema de Transporte

Los resultados tanto de la modulación como de la demodulación por parte de las tarjetas realizadas fue la clave de un Sistema de Transporte exitoso. A continuación en la figura 47 se observan gráficas vistas desde un osciloscopio a la salida del modulador y demodulador al momento de transmitir.

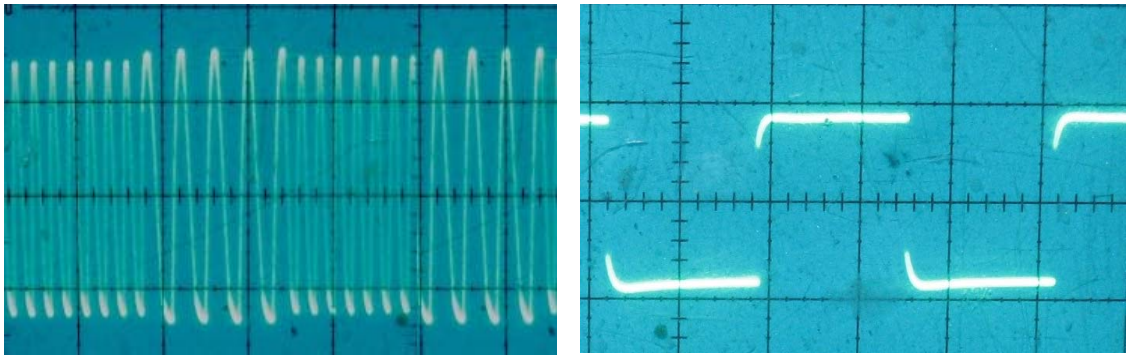


Figura 47. a) Señal Modulada; b) Señal demodulada.

Para la elaboración de las tarjetas de transporte se procedió de la misma manera que la Tarjeta de Adquisición y Control de Datos. A continuación se presentan en las figuras 48, 49 y 50 los circuitos impresos de la Estación Central y el acabado final de la misma.

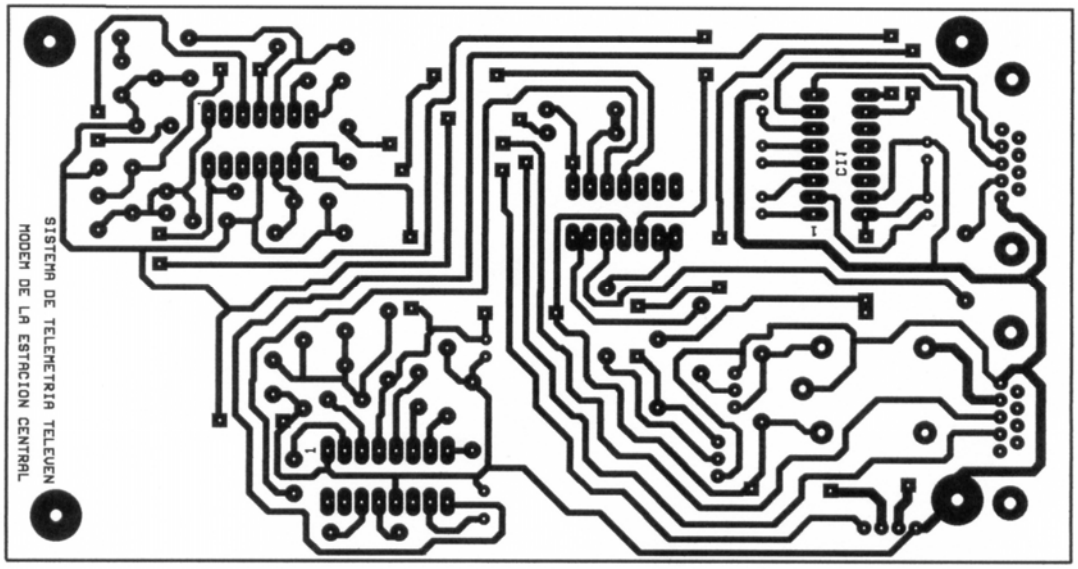


Figura 48. Circuito Impreso de la Tarjeta Estación Central.

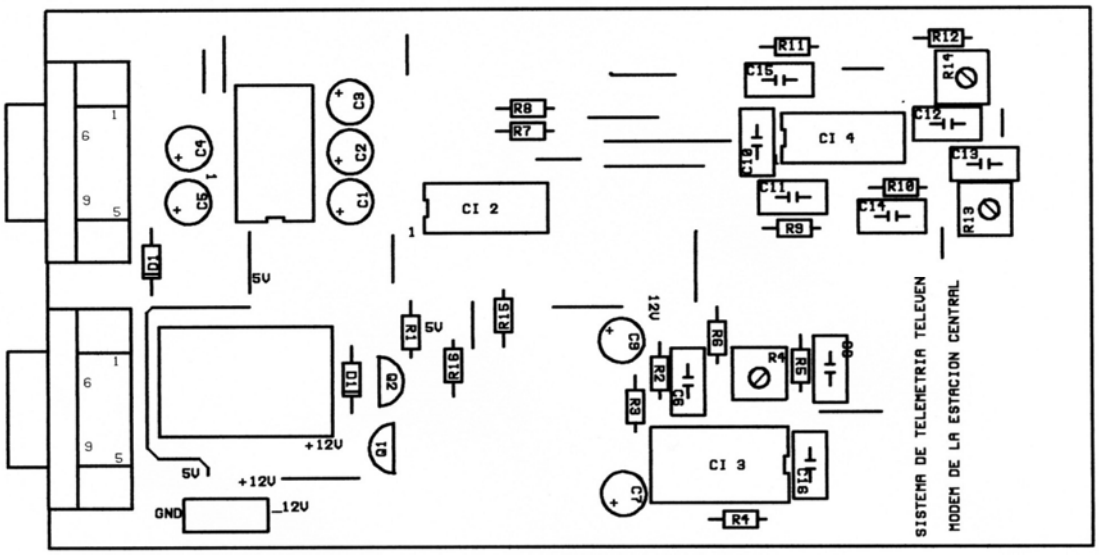


Figura 49. Distribución de la Tarjeta Estación Central.

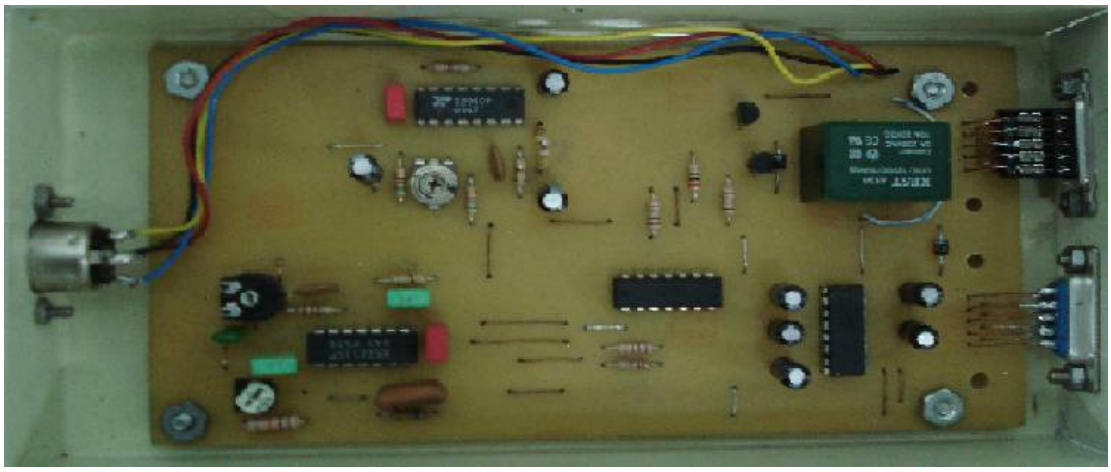


Figura 50. Acabado final de la Tarjeta de Transporte Central.

La Tarjeta de la Estación Central como se puede observar en las anteriores figuras consta de tres salidas, un conector DB-9 Hembra que va conectado al puerto COM-1 de la PC, un conector DB-9 Macho dirigido hacia la conexión al radio, y un conector para la alimentación del circuito, el cual utiliza una fuente de poder conmutada AC-DC Modelo FPGP36 (Anexo 4) y que proporciona salidas de 5V, 12V, -12V.

De la misma manera la Tarjeta MODEM de la Estación Repetidora posee tres salidas, de las cuales dos de ellas van dirigidas hacia la Tarjeta de Adquisición y Control de Datos, ya que estas dos tarjetas en conjunto forman el Sistema completo para una Estación Repetidora. Estas dos salidas son la de la alimentación para el circuito y la otra para la Transmisión, Recepción y PTT; la tercera salida faltante es la del conector DB-9 Macho que va dirigido hacia la conexión al radio.

Las pruebas finales de petición de reportes y control de transmisión arrojaron los siguientes resultados: de 20 reportes solicitados 17 fueron completados dando un 85% de eficiencia, y de los 17 reportes el 100% de la data era la correcta.

A continuación se presentan en las figuras 51, 52 y 53 los circuitos impresos de la Estación Repetidora y el acabado final de la misma.

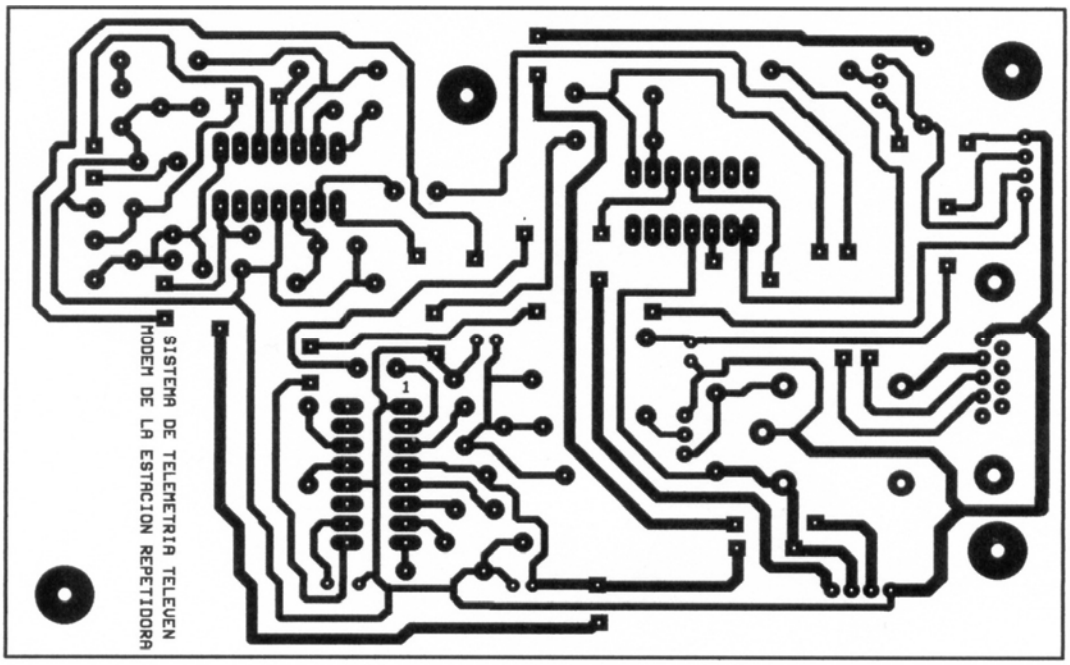


Figura 51. Circuito Impreso de la Tarjeta Estación Repetidora.

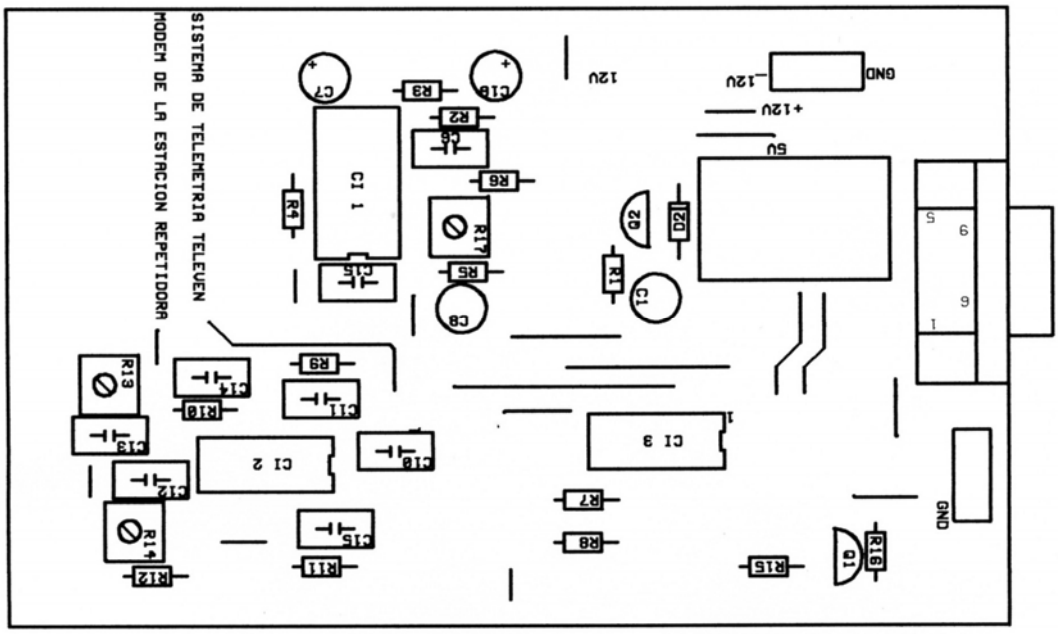


Figura 52. Distribución de la Tarjeta Estación Repetidora.

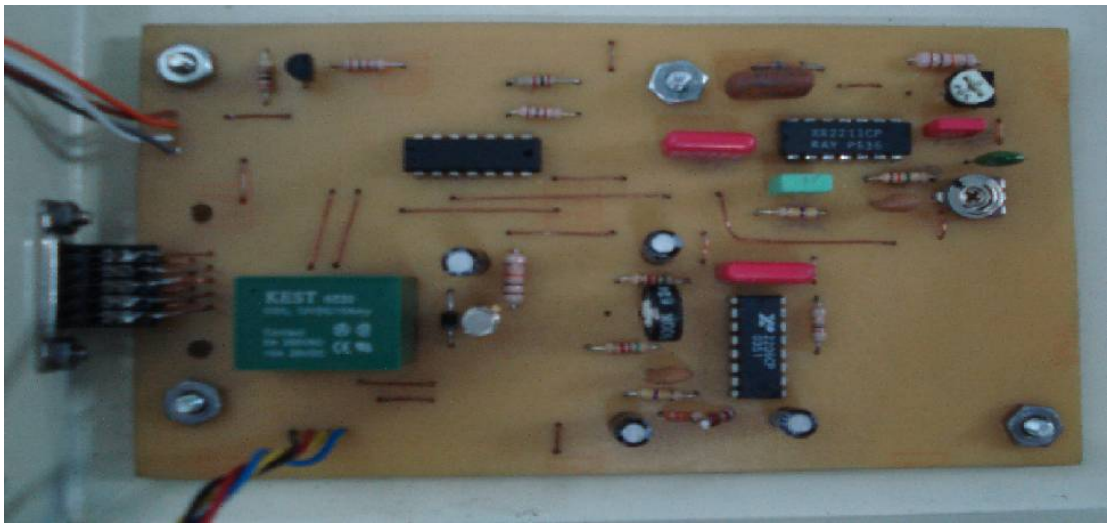


Figura 53. Tarjeta de Transporte Repetidora.

A continuación se presenta en la figura 54 tanto el Hardware ubicado en la Estación Central (Caja pequeña), como el ubicado en la Estación Repetidora de Caricuao (Caja Grande).

Figura 54. Acabado final del Hardware del Sistema de Telemetría.

Capítulo V

Conclusiones y Recomendaciones

• Conclusiones

- La Telemetría es una tecnología con la capacidad de integración electrónica y humana, la cual permite monitorear y controlar distintos parámetros en forma remota, haciendo un sistema muy versátil y eficaz, ya que ahorra tiempo, traslado, comodidad, etc.

- La sencillez de la interfaz del software de Monitoreo y Control en cuanto a manejo y la forma gráfica en la que se presenta la información, hace de éste una herramienta bastante amigable al operador de la Central que se interese en conocer el estado de operatividad de alguna estación.

- El monitoreo de los distintos parámetros permite una supervisión constante de los equipos de la Estación Repetidora, con la ventaja de mantener un registro de todos los reportes realizados.

- El control del Apagado/Encendido del Transmisor proporciona al operador comodidad, seguridad y un sustentable ahorro de tiempo, ya que se evita el traslado hacia la estación repetidora en los casos en que no haya personal en la misma.

- La Base de Datos implementada permite poseer un registro mensual de todos los reportes realizados por el operador, dando a la empresa un historial del comportamiento de sus repetidoras.

- Las gráficas dinámicas proyectadas por la Base de Datos ofrecen un comportamiento mensual de los parámetros más importantes, concediendo al operador facilidad de análisis.

- El PIC 16F877 constituye la herramienta fundamental de la Tarjeta de Adquisición y Control de Datos, ya que por sus múltiples entradas y salidas digitales y analógicas permite recoger gran cantidad de parámetros para tener una idea clara y completa del estado de operatividad de la estación.

- La simpleza del diseño del hardware de transporte (modulador y demodulador FSK) y su buen funcionamiento hacen de éste un sistema económico, sencillo y confiable.

- La modulación FSK presenta mayores ventajas sobre la modulación OOK, siendo la mas importante la inmunidad al ruido y las interferencias, lo cual es de gran valor para la transmisión de data vía radio.

• Recomendaciones

Como futura recomendación para este proyecto tan ambicioso como lo es un Sistema de Telemetría para el Monitoreo y Control de las Estaciones Repetidoras de Televen, se llegó a la conclusión de sugerir ciertas mejoras al diseño elaborado durante este Trabajo Especial de Grado.

Una de las mejoras que permitiría una mayor eficiencia para el envío y recepción de la data al momento de querer magnificar el proyecto con todas las repetidoras a nivel nacional sería tener un canal de radio-frecuencia exclusivo para la Telemetría, en el cual no existan interferencias ajenas a la data en cuestión. Actualmente el

sistema se implementó con los radios UHF existentes en Televen, en donde el canal es utilizado para las comunicaciones entre los operadores de las estaciones.

Otra mejora necesaria para optimizar el Sistema de Telemetría es la comunicación automática del estado de cada uno de los parámetros al ocurrir cierto evento, por ejemplo, si en algún momento el transmisor de alguna Estación Repetidora se apaga por alguna circunstancia, es necesario enviar un señal de alerta a la Estación Central informando el suceso ocurrido, y así poder interferir en el problema con la mayor rapidez posible; de la misma manera que el transmisor se puede realizar con la temperatura de la Sala en donde si existe una variación fuera de los rangos normales envíe una alerta con lo sucedido.

Por último, el Sistema de Telemetría realizado solo es el comienzo de muchas de las funciones que se pueden lograr, la ampliación en el monitoreo de parámetros puede ser considerablemente mayor, se pueden incluir funciones como encendido/apagado de la planta eléctrica, control del aire acondicionado, entre muchos otros.

Bibliografía

- Haykin (2002). Sistemas de Comunicación. *Sistemas de Modulación*. Editorial Limusa S.A. Páginas:109 – 115.

- Microchip Inc (2001). PIC16F87X Data Sheet 28/40-Pin 8-Bit CMOS Flash Microcontrollers.Microchip Technology Incorporated. Extraído el 25 de noviembre de 2005. http://www.microchip.com /downloads/en/DeviceDoc/30292c.pdf. Páginas: 136-142.

- Technicothica (2004). Tone Generator and Tone Decoder Integrated Circuits. Extraído el 30 de febrero de 2006. http://www.klm-tech.com/technicothica/xr.html.

- Exar Corporation (Junio de 1997). XR-2206 Monolithic Function Generator. Extraído el 20 de junio de 2006. http://www.exar.com/products/XR2206v103.pdf#search=%22xr2206%22.

- Exar Corporation (Junio de 1997). XR-2211 FSK Demodulator/ Tone Coder. Extraído el 20 de junio de 2006. http://www.exar.com/products/XR2211v301.pdf.

- National Semiconductor (Noviembre de 2000). LM135/LM235/LM335, LM135A/LM235A/LM335A Precision Temperature Sensors. Extraído el 27 de Agosto de 2006 http://www.national.com/ds/LM/LM135.pdf.

- Bloger Electrónica (Septiembre 10 de 2006)**.** Progamación de PICs en lenguaje C. Extraído el 20 de junio de 2006http://loslocoselectro.blogspot.com/.

- Maxim IC (Febrero de 2003). Maxim Multichannel RS-232 Driver. Extraído el 27 de mayo de 2006.http://www.maxim-ic.com/quick_view2.cfm/qv_pk/1798.

- SGS Thomson Microelectronics (Octubre de 2005). General Purpose Quad J-FET Operational Amplifiers. Extraído el 20 de junio de 2006. http://www.datasheetcatalog.net/es/datasheets_pdf/T/L/0/8/TL084.shtml.

- De Rueda, Luis. Microcontroladores - Tutorial MPLab Ide. Ero-Pic. 2003. Extraído el 06 de marzo de 2006. http://perso.wanadoo.es/luis_ju/pic2/ind_pic2.html.

- García de Jalón, Javier y Rodríguez, José Ignacio Rodríguez. (Agosto de 1999). Visual Basic 6.0. Escuela Superior de Ingenieros Industriales. Universidad Navarra.

- Bailey, David (2002). Practical Radio Engineering and Telemetry for Industry. Newnes.

Apéndice A

Planos electrónicos de la Tarjeta de Adquisición y Control de Datos

Apéndice B

Planos Electrónicos de la Tarjeta Modem de la Estación Central

Apéndice C

Planos Electrónicos de la Tarjeta Modem de la Estación Repetidora

Apéndice D

Código Visual Basic del Sistema de Monitoreo y Control

A continuación se presenta el código del Software de Monitoreo y Control debido a la extensión del mismo y a la constante repetición de líneas y funciones se optó por resumir las funciones principales, y para un solo estado (Distrito Federal) ya que para los demás estados es el mismo procedimiento.

Form Principal.-

```
Option Explicit
Private Declare Function SetWindowLong Lib "user32" Alias "SetWindowLongA" (ByVal hwnd As Long, ByVal nIndex As Long, ByVal dwNewLong As Long) As Long
Private Declare Function GetWindowLong Lib "user32" Alias "GetWindowLongA" (ByVal hwnd As Long, ByVal nIndex As Long) As Long
Private Declare Function DeleteMenu Lib "user32" (ByVal hMenu As Long, ByVal nPosition As Long, ByVal wFlags As Long) As Long
Private Declare Function GetSystemMenu Lib "user32" (ByVal hwnd As Long, ByVal bRevert As Long) As Long
Public salida, enviado, LedB, LedC, Verde, Rojo, mybox, enviado3 As String
Public error, i, j, k, AD0, AD1, AD3, StatusPORT As Double
Public mybox1, id As String
Public Timer2Counter As Integer

Private Sub Base_Click()
fbase.Visible = True
End Sub

Private Sub Form_Load()
incluye_salto_carro = Not (incluye_salto_carro)
incluye_salto_carro = True
MSComm1.InputLen = 0
MSComm1.RThreshold = 1
MSComm1.SThreshold = 1
StatusPORT = 0
MSComm1.RTSEnable = False
MSComm1.DTREnable = False
Logo.Visible = True
Salir.Visible = True
Venezuela.Visible = True
mapa.Visible = True
Tiempo.Caption = Time$
date.Caption = DateTime.date$
End Sub
```

```
Public Sub apuerto_Click()
On Error GoTo manejar_errores
  MSComm1.CommPort = 2
  MSComm1.Settings = "300" + ",N,8,1"
  MSComm1.PortOpen = True
  StatusPORT = 1
  GoTo Salir
manejar_errores:
 mybox = MsgBox("Error detectado por Visual Basic: " + Err.Description, vbCritical, "Error")
 Resume Salir
Salir:
End Sub

Public Sub Enviar_Click()
    On Error GoTo manejar_errores
    MSComm1.DTREnable = True
    Timer3.Interval = 800
    Timer1.Interval = 100
    Timer2.Interval = 1000
    Timer4.Interval = 1100
    Timer2Counter = 0
    Timer3.Enabled = True
    GoTo Salir
manejar_errores:
    mybox = MsgBox("Error detectado por Visual Basic: " + Err.Description, vbCritical, "Error")
    MSComm1.DTREnable = False
    Call Back_Click
    Resume Salir
Salir:
End Sub

Private Sub PTT()
    On Error GoTo manejar_errores
    Timer6.Enabled = True
    MSComm1.Output = id
    Timer4.Enabled = True
    Timer1.Interval = 100
    Timer2.Interval = 5000
    Timer2Counter = 0
    Timer1.Enabled = True
    Timer2.Enabled = True
    GoTo Salir
manejar_errores:
    mybox = MsgBox("Error detectado por Visual Basic: " + Err.Description, vbCritical, "Error")
    Call Back_Click
    Resume Salir
Salir:
End Sub

Private Sub CPuerto_Click() ' Cierra el Puerto
On Error GoTo manejar_errores
 MSComm1.PortOpen = False
```

```
 StatusPORT = 0
 GoTo Salir
manejar_errores:
 mybox = MsgBox("Error detectado por Visual Basic: " + Err.Description, vbCritical, "Error")
 Resume Salir
Salir:
End Sub

Private Sub MSComm1_OnComm()
Dim envio As Boolean
envio = False
If MSComm1.CommEvent = comEvReceive Then
   enviado = MSComm1.Input
End If
End Sub

Private Sub Timer1_Timer()
      Espera.Visible = True
      If Mid(enviado, 1, 2) = "UA" Then
      enviado = Mid(enviado, 3, 4)
      Espera.Visible = False
      Timer2.Enabled = False
      Timer1.Enabled = False
      MSComm1.DTREnable = False
      mybox = MsgBox("Reporte completado con exito", vbInformation, "Información")
      Call VReporte_Click
      End If
End Sub

Private Sub Timer2_Timer()
   Timer2Counter = Timer2Counter + 1
      If Timer2Counter = 3 Then
      Espera.Visible = False
      MSComm1.DTREnable = False
      Timer1.Enabled = False
      Timer2.Enabled = False
      mybox = MsgBox("El Reporte ha sido fallido", vbCritical, "Advertencia")
      Call Back_Click
   End If
End Sub

Private Sub Timer3_Timer() ' Timer para llamar a PTT
   Timer3.Enabled = False
   Call PTT
End Sub

Private Sub Timer4_Timer() ' Timer de duración del PTT
   MSComm1.DTREnable = False
   Timer4.Enabled = False
End Sub
```

```
Private Sub VReporte_Click()   ' Muestra el Form3 que es el de los reportes
   Form3.Show vbModal, Me
Call Back_Click
End Sub

Private Sub Mapa_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
frame1.Caption = "Estaciones"
Label1.Caption = " "
Label2.Caption = " "
Label3.Caption = " "
Label4.Caption = ""
Label5.Caption = ""
Distrito.Visible = False
End Sub

Private Sub ldistrito_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
Distrito.Visible = True
frame1.Caption = "Distrito Federal"
Label1.Caption = " - Caricuao"
Label2.Caption = " - El cuño"
End Sub

Private Sub ldistrito_Click()
   fdistrito.Visible = True
   fvenezuela.Visible = False
End Sub

Private Sub Salir_Click()
Me.Hide
End
End Sub

Private Sub Back_Click()
   Call CPuerto_Click
End Sub
```

Form Reporte.-

```
Private Sub Form_Load()
   Dim b As Integer
   enviado2 = fvenezuela.enviado
   fvenezuela.enviado = ""
   fvenezuela.enviado3 = enviado2
   Text1.Text = enviado2
   LedB = Dec2Bin(Mid(enviado2, 1, 1))
   For b = 1 To 8
      If Mid(LedB, b, 1) = "1" Then
         Shape1(b - 1).BackColor = &HFF00&
         Else
         Shape1(b - 1).BackColor = &HFF&
      End If
   Next
   LedC = Dec2Bin(Mid(enviado2, 2, 1))
```

```
   For b = 2 To 7
     If Mid(LedC, b, 1) = "1" Then
        Shape1((b + 7) - 1).BackColor = &HFF00&
        Else
        Shape1((b + 7) - 1).BackColor = &HFF&
      End If
   Next
  AD0 = Asc(Mid(enviado2, 3, 1)) * 256 + Asc(Mid(enviado2, 4, 1))
  Label17(0).Caption = Mid(Str$(AD0 * 5 / 1023), 2) + "%"
  AD1 = Asc(Mid(enviado2, 5, 1)) * 256 + Asc(Mid(enviado2, 6, 1))
   Label17(1).Caption = Mid(Str$(AD1 * 5 / 1023), 2) + "%"
   AD3 = Asc(Mid(enviado2, 3, 1)) * 256 + Asc(Mid(enviado2, 4, 1))
   Label17(2).Caption = Mid(Mid(Str$(100 * (AD3 * 5 / 1023) - 273), 2), 1, 4) + " C "
   Form3.Caracteristicas.Visible = False
   Timer1.Enabled = True
   Encender.Enabled = False
   Apagar.Enabled = False
   Shape3.FillColor = Shape1(6).BackColor
   If Mid(fvenezuela.id, 4, 1) = "L" Then
   Estado.Caption = "Distrito federal"
   Estacion.Caption = "Caricuao"
   Hora.Caption = Time$
   Fecha.Caption = date$
   End If
End Sub

Private Sub Guardar_Click()
If Mid(fvenezuela.id, 4, 1) = "L" Then
Call caricuao_Click
Set xlLibro = xlsApp.Workbooks(1)
xlLibro.Save
xlsApp.Workbooks(1).Close
Set xlsApp = Nothing
Guardar.Enabled = False
mybox = MsgBox("Su Reporte ha sido guardado en:" + xlspath, vbInformation, "Guardado")
End If
End Sub

Private Sub caricuao_Click()
xlspath = "C:\Sistema Telemetria\Reportes\Distrito Federal\Caricuao.xls"
Set xlsApp = New Excel.Application
xlsApp.Visible = False
xlsApp.Workbooks.Open xlspath
mes = Mid(date$, 1, 2)
If mes = "01" Then
a = xlsApp.Worksheets("Enero").Cells(1, 35) + 1
Set xlHoja1 = xlsApp.Worksheets("Enero")
With xlHoja1
.Cells(2, a + 1) = date$
.Cells(3, a + 1) = Time$
.Cells(4, a + 1) = Mid(Form3.LedC, 1, 1)
.Cells(5, a + 1) = Mid(Form3.LedC, 2, 1)
.Cells(6, a + 1) = Mid(Form3.LedC, 3, 1)
```

```
.Cells(7, a + 1) = Mid(Form3.LedC, 4, 1)
.Cells(8, a + 1) = Mid(Form3.LedC, 5, 1)
.Cells(9, a + 1) = Mid(Form3.LedC, 6, 1)
.Cells(10, a + 1) = Mid(Form3.LedC, 7, 1)
.Cells(11, a + 1) = Mid(Form3.LedC, 8, 1)
.Cells(12, a + 1) = Mid(Form3.Label17(0).Caption, 1, 8)
.Cells(13, a + 1) = Mid(Mid(Form3.Label17(1).Caption, 1, 8).Caption, 1, 8)
.Cells(14, a + 1) = Mid(Form3.Label17(2).Caption, 1, 8)
.Cells(1, 35) = a
End With
End If
End Sub

Private Sub Aceptar_Click()
If Estados.Text = "Distrito Federal" Then
If Distrito.Text = "El Cuño" Then
Set xlsApp = New Excel.Application
xlsApp.Visible = True
xlsApp.Workbooks.Open "C:\Sistema Telemetria\Reportes\Distrito Federal\El Cuño.xls"
End If
If Distrito.Text = "Caricuao" Then
Set xlsApp = New Excel.Application
xlsApp.Visible = True
xlsApp.Workbooks.Open "C:\Sistema Telemetria\Reportes\Distrito Federal\Caricuao.xls"
End If
End If
End Sub
```

Form Estado Distrito Federal.-

```
Private Sub Form_Load()
caricuao.Visible = False
cuño.Visible = False
lregresar.Visible = False
End Sub

Private Sub acaricuao_Click()
fvenezuela.id = "UUUL□]"
fvenezuela.apuerto_Click
fvenezuela.Enviar_Click
End Sub

Private Sub acuño_Click()
fvenezuela.id = " UUUM□]"
fvenezuela.apuerto_Click
fvenezuela.Enviar_Click
End Sub

Private Sub Back_Click()
Me.Hide
fvenezuela.Visible = True
End Sub
```

```
Private Sub acaricuao_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
caricuao.Visible = True
Latitud.Caption = "10° 26' 46""
Longitud.Caption = "66° 58' 47""
End Sub

Private Sub acuño_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
cuño.Visible = True
Latitud.Caption = "10° 31' 12""
Longitud.Caption = "66° 52' 50""
End Sub

Private Sub Mapa_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
caricuao.Visible = False
cuño.Visible = False
Longitud.Caption = " "
Latitud.Caption = " "
End Sub

Private Sub Timer6_Timer()
Timer6.Enabled = False
fvenezuela.id = "UUUL^"
fvenezuela.apuerto_Click
fvenezuela.Enviar_Click
End Sub

Private Sub Timer7_Timer()
Timer7.Enabled = False
fvenezuela.id = "UUUL-_"
fvenezuela.apuerto_Click
fvenezuela.Enviar_Click
End Sub
```

Apéndice E

Código del Microcontrolador PIC16F877

```
LIST    P=16F877

        RADIX HEX

        INCLUDE <P16F877.inc>

CONV_HIGH   EQU 20h
CONV_LOW    EQU 21h
CounterA    EQU 22h
CounterB    EQU 23h
BUFFER      EQU 24h
R_ContA     EQU 25h
R_ContB     EQU 26h
R_ContC     EQU 27h
Reset       EQU 28h
ID          EQU 29h
CHK         EQU 2Ah
PAY         EQU 2Bh
PAY2        EQU 2Ch

        ORG 0x00
        goto RS232

        ORG 0x04
        goto INTERR

RS232
        call BANCO0
        clrf PORTB
        clrf PORTA
        clrf PORTD
        clrf PORTC
        call BANCO1
        movlw b'00001011'
        movwf TRISA
        movlw b'11111111'
        movwf TRISB
        movlw b'10111111'
        movwf TRISC
        movlw b'00000000'
        movwf TRISD
        movlw b'00100000'
        movwf TXSTA
        movlw .207
        movwf PIE1
        call BANCO0
        movlw b'10010000'
        movwf RCSTA
        clrf RCREG
        movlw b'11000000'
        movwf INTCON
        movlw b'00000000'
        movwf Reset
        goto ESPERA
BANCO1
        bsf STATUS,RP0
        bcf STATUS,RP1
        return
BANCO0
        bcf STATUS,RP0
        bcf STATUS,RP1
        return
ESPERA
        ;btfss INTCON,GIE
        ;goto RS232
        goto ESPERA
INTERR
        ;call BANCO1
        ;bcf INTCON,GIE
        call BANCO0
        btfss PIR1,RCIF
        goto EXIT_INTERR
        clrf BUFFER
        movf RCREG,W
        movwf BUFFER
        xorlw 'U'
        btfss STATUS,Z
        goto Seguir
        goto Listo2
Seguir
        movf Reset,0
        xorlw d'0'
        btfss STATUS,Z
        goto Listo
        goto EXIT_INTERR
Listo2
        movlw d'0'
        movwf Reset
```

```
Listo
        movlw   d'0'
        addlw   d'1'
        addwf   Reset,1
        movf    Reset,0
        xorlw   d'4'
        btfss   STATUS,Z
        goto    Verificar
        goto    Analizar

        ;SKIP iF SET --> z=1

Verificar
        movf    Reset,0
        xorlw   d'1'
        btfss   STATUS,Z
        goto    Verificar2
        goto    EXIT_INTERR

Verificar2
        movf    Reset,0
        xorlw   d'2'
        btfss   STATUS,Z
        goto    Guardar3
        goto    Guardar2

Guardar2
        clrf    ID
        movf    RCREG,W

        movwf   ID
        goto    EXIT_INTERR

Guardar3
        clrf    PAY
        movf    RCREG,W
        movwf   PAY
        goto    EXIT_INTERR
Analizar
        clrf    CHK
        movf    RCREG,W

        movwf   CHK
        movlw   d'0'
        movwf   Reset
        ;goto   EXIT_INTERR
        movf    PAY,0
        movwf   PAY2
        movf    ID,0
        addwf   PAY2,1
        movf    CHK,0
        xorwf   PAY2,0
        btfss   STATUS,Z
        goto    EXIT_INTERR
        ;bsf    PORTD,3
        ;call   Retardo_1s
        ;bcf    PORTD,3
        goto    COMPARAR

EXIT_INTERR
        retfie
        ;goto INTERR

COMPARAR
        movf    ID,W
        xorlw   'A'
        btfss   STATUS,Z
        goto    EXIT_INTERR
        goto    COMPARAR2

COMPARAR2
        movf    PAY,W
        xorlw   d'28'
        btfss   STATUS,Z
        goto    COMPARAR3
        goto    ENVIAR_U
COMPARAR3
        movf    PAY,W
        xorlw   d'29'
        btfss   STATUS,Z
        goto    COMPARAR4
        goto    Encender
COMPARAR4
        movf    PAY,W
        xorlw   d'30'
        btfss   STATUS,Z
        goto    EXIT_INTERR
        goto    Apagar
Encender
        bsf     PORTD,7
        call    Retardo_1s
        bcf     PORTD,7
        call    Retardo_1s
        bsf     PORTD,5
        call    Retardo_1s
        bcf     PORTD,5
        call    Retardo_1s
        bsf     PORTD,6
        call    Retardo_1s
        bcf     PORTD,6
        call    ENVIAR_U
        goto    EXIT_INTERR
```

```
Apagar
        bsf     PORTD,7
        call    Retardo_1s
        bcf     PORTD,7
        call    Retardo_1s
        bsf     PORTD,4
        call    Retardo_1s
        bcf     PORTD,4
        call    Retardo_1s
        bsf     PORTD,6
        call    Retardo_1s
        bcf     PORTD,6
        call    ENVIAR_U
        goto    EXIT_INTERR
ENVIAR_U
        call    Retardo_1s
        bsf     PORTD,3         ;PTT
        call    Retardo_1s
        movlw   'U'
        movwf   TXREG
        call    BANCO1
TX?_U
        btfss   TXSTA,TRMT
        goto    TX?_U
        call    BANCO0
        call    DELAY
        goto    ENVIAR_A
ENVIAR_A
        movlw   'A'
        movwf   TXREG
        call    BANCO1
TX?_A
        btfss   TXSTA,TRMT
        goto    TX?_A
        call    BANCO0
        call    DELAY
        goto    ENVIAR_B
ENVIAR_B
        movf    PORTB,W
        movwf   TXREG
        call    BANCO1
TX?_B
        btfss   TXSTA,TRMT
        goto    TX?_B
        call    BANCO0
        call    DELAY
        goto    ENVIAR_C
ENVIAR_C
        movf    PORTC,W
        movwf   TXREG
        call    BANCO1
TX?_C
        btfss   TXSTA,TRMT
        goto    TX?_C
        call    BANCO0
        call    DELAY
        goto    AD_AN3
AD_AN0
        movlw   b'10000001'
        movwf   ADCON0
        bcf     PIR1,ADIF
        call    BANCO1
        movlw   b'10000100'
        movwf   ADCON1
        call    BANCO0
        bsf     ADCON0,GO
AD?_AN0
        btfss   PIR1,ADIF
        goto    AD?_AN0
ENVIAR_AN0
        bcf     PIR1,ADIF
        movf    ADRESH,W
        movwf   CONV_HIGH
        call    BANCO1
        movf    ADRESL,W
        call    BANCO0
        movwf   CONV_LOW
        movf    CONV_HIGH,W
        movwf   TXREG
        call    BANCO1
TX?_AN0_HIGH
        btfss   TXSTA,TRMT
        goto    TX?_AN0_HIGH
        call    BANCO0
        call    DELAY
        movf    CONV_LOW,W
        movwf   TXREG
        call    BANCO1
TX?_AN0_LOW
        btfss   TXSTA,TRMT
        goto    TX?_AN0_LOW

        call    BANCO0
        call    DELAY
        goto    AD_AN1

AD_AN1
        movlw   b'10001001'
        movwf   ADCON0

        bcf     PIR1,ADIF
```

```
                call    BANCO1
                movlw   b'10000100'
                movwf   ADCON1
                call    BANCO0
                bsf     ADCON0,GO
AD?_AN1
                btfss   PIR1,ADIF
                goto    AD?_AN1
ENVIAR_AN1
                bcf     PIR1,ADIF
                movf    ADRESH,W
                movwf   CONV_HIGH
                call    BANCO1
                movf    ADRESL,W
                call    BANCO0
                movwf   CONV_LOW
                movf    CONV_HIGH,W
                movwf   TXREG
                call    BANCO1
TX?_AN1_HIGH
                btfss   TXSTA,TRMT
                goto    TX?_AN1_HIGH
                call    BANCO0
                call    DELAY
                movf    CONV_LOW,W
                movwf   TXREG
                call    BANCO1
TX?_AN1_LOW
                btfss   TXSTA,TRMT
                goto    TX?_AN1_LOW
                call    BANCO0
                call    DELAY
                goto    AD_AN3
AD_AN3
                movlw   b'10011001'
                movwf   ADCON0
                bcf     PIR1,ADIF
                call    BANCO1
                movlw   b'10000100'
                movwf   ADCON1
                call    BANCO0
                bsf     ADCON0,GO
                btfss   PIR1,ADIF
                goto    AD?_AN3
ENVIAR_AN3
                bcf     PIR1,ADIF
                movf    ADRESH,W
                movwf   CONV_HIGH
                call    BANCO1
                movf    ADRESL,W
                call    BANCO0
                movwf   CONV_LOW
                movf    CONV_HIGH,W
                movwf   TXREG
                call    BANCO1
TX?_AN3_HIGH
                btfss   TXSTA,TRMT
                goto    TX?_AN3_HIGH
                call    BANCO0
                call    DELAY
                movf    CONV_LOW,W
                movwf   TXREG
                call    BANCO1
TX?_AN3_LOW
                btfss   TXSTA,TRMT
                goto    TX?_AN3_LOW
                call    BANCO0
                call    DELAY
                bcf     PORTD,3
                call    Retardo_500ms
                bsf     PORTD,0
                call    Retardo_1s
                bcf     PORTD,0
                goto    EXIT_INTERR
DELAY
                return
                movlw   D'255'
                movwf   CounterA
                movlw   D'255'
                movwf   CounterB
LOOP1
                nop
                nop
                nop
                nop
                decfsz  CounterA,1
                goto    LOOP1
                movlw   D'255'
                movwf   CounterA
LOOP2
                nop
                nop
                nop
                nop
                decfsz  CounterA,1
                goto    LOOP2
                decfsz  CounterB,1
                goto    LOOP1
                return
; RETARDOS de 0.5 hasta 20 segundos
Retardo_20s
        movlw   d'200'
        goto    Retardo_1Decima
```

```
Retardo_10s
        movlw   d'100'

goto    Retardo_1Decima
Retardo_5s
        movlw   d'50'
        goto    Retardo_1Decima
Retardo_2s
        movlw   d'20'
        goto    Retardo_1Decima
Retardo_1s
        movlw   d'10'
        goto    Retardo_1Decima
Retardo_500ms
        movlw   d'5'

Retardo_1Decima
        movwf   R_ContC
R1Decima_BucleExterno2
        movlw   d'100'
        movwf   R_ContB
R1Decima_BucleExterno
        movlw   d'249'
        movwf   R_ContA
R1Decima_BucleInterno
        nop
        decfsz  R_ContA,F
        goto    R1Decima_BucleInterno
        decfsz  R_ContB,F
        goto    R1Decima_BucleExterno
        decfsz  R_ContC,F
        goto    R1Decima_BucleExterno2
        return

END
```

Anexo A

Especificaciones técnicas del PIC16F877

PIC16F87X

28/40-pin 8-Bit CMOS FLASH Microcontrollers

Devices Included in this Data Sheet:

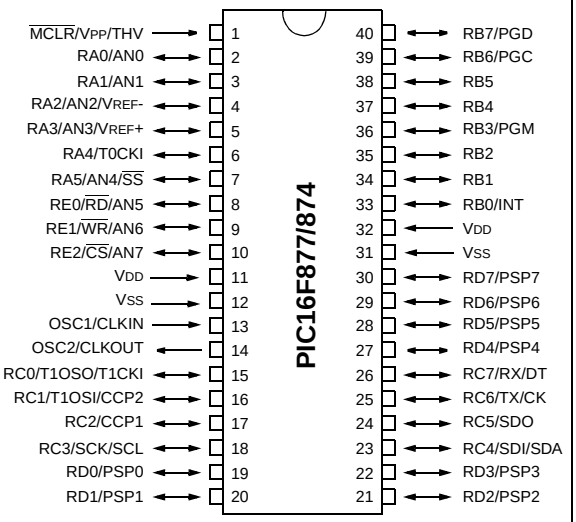
- PIC16F873
- PIC16F874
- PIC16F876
- PIC16F877

Microcontroller Core Features:

- High-performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input
  DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory,
  Up to 368 x 8 bytes of Data Memory (RAM)
  Up to 256 x 8 bytes of EEPROM data memory
- Pinout compatible to the PIC16C73B/74B/76/77
- Interrupt capability (up to 14 sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and
  Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code-protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low-power, high-speed CMOS FLASH/EEPROM technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP) via two pins
- Single 5V In-Circuit Serial Programming capability
- In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial and Industrial temperature ranges
- Low-power consumption:
  - < 2 mA typical @ 5V, 4 MHz
  - 20 μA typical @ 3V, 32 kHz
  - < 1 μA typical standby current

Pin Diagram

PDIP



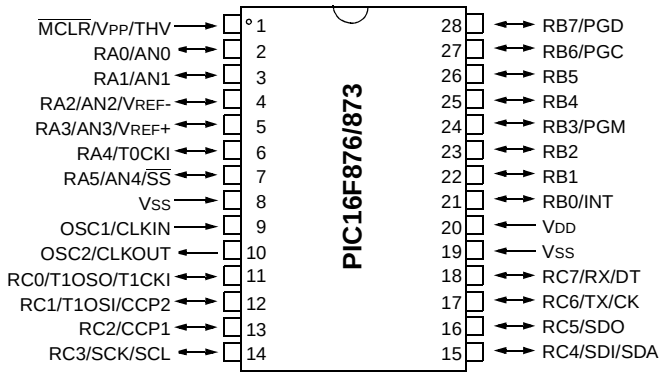
Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules
  - Capture is 16-bit, max. resolution is 12.5 ns
  - Compare is 16-bit, max. resolution is 200 ns
  - PWM max. resolution is 10-bit
- 10-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with SPI™ (Master Mode) and I^2C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) 8-bits wide, with external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)

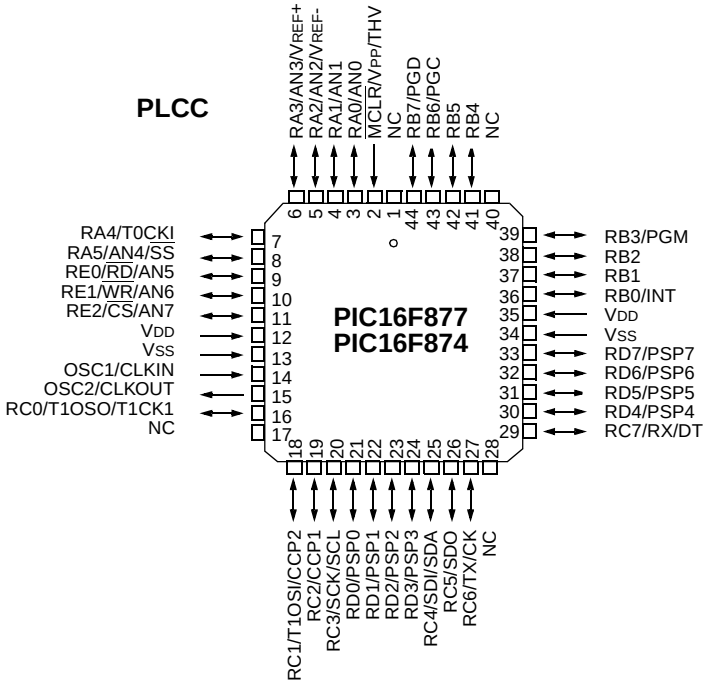
© 1999 Microchip Technology Inc.

Pin Diagrams

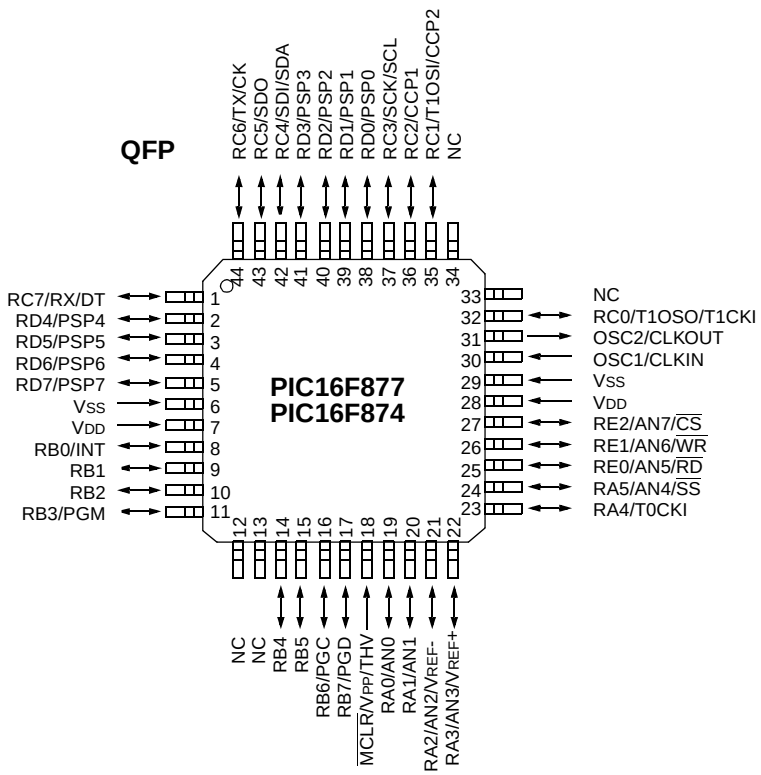
DIP, SOIC



PLCC



QFP



© 1999 Microchip Technology Inc.

Key Features PICmicro™ Mid-Range Reference Manual (DS33023)	PIC16F873	PIC16F874	PIC16F876	PIC16F877
Operating Frequency	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz
Resets (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
FLASH Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory	128	128	256	256
Interrupts	13	14	13	14
I/O Ports	Ports A,B,C	Ports A,B,C,D,E	Ports A,B,C	Ports A,B,C,D,E
Timers	3	3	3	3
Capture/Compare/PWM modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Instruction Set	35 Instructions	35 Instructions	35 Instructions	35 Instructions

© 1999 Microchip Technology Inc.

Table of Contents

1.0 Device Overview 5
2.0 Memory Organization 11
3.0 I/O Ports 29
4.0 Data EEPROM and FLASH Program Memory 41
5.0 Timer0 Module 47
6.0 Timer1 Module 51
7.0 Timer2 Module 55
8.0 Capture/Compare/PWM (CCP) Module(s) 57
9.0 Master Synchronous Serial Port (MSSP) Module 63
10.0 Universal Synchronous Asynchronous Receiver Transmitter (USART) 95
11.0 Analog-to-Digital Converter (A/D) Module 111
12.0 Special Features of the CPU 121
13.0 Instruction Set Summary 137
14.0 Development Support 145
15.0 Electrical Characteristics 151
16.0 DC and AC Characteristics Graphs and Tables 173
17.0 Packaging Information 175
Appendix A: Revision History 183
Appendix B: Device Differences 183
Appendix C: Conversion Considerations 183
Index 185
On-Line Support 191
Product Identification System 193

To Our Valued Customers

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number. e.g., DS30000A is version A of document DS30000.

New Customer Notification System

Register on our web site (www.microchip.com/cn) to receive the most current information on our products.

Errata

An errata sheet may exist for current devices, describing minor operational differences (from the data sheet) and recommended workarounds. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; http://www.microchip.com
- Your local Microchip sales office (see last page)
- The Microchip Corporate Literature Center; U.S. FAX: (480) 786-7277

When contacting a sales office or the literature center, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Corrections to this Data Sheet

We constantly strive to improve the quality of all our products and documentation. We have spent a great deal of time to ensure that this document is correct. However, we realize that we may have missed a few things. If you find any information that is missing or appears in error, please:

- Fill out and mail in the reader response form in the back of this data sheet.
- E-mail us at webmaster@microchip.com.

We appreciate your assistance in making this a better document.

© 1999 Microchip Technology Inc.

1.0 DEVICE OVERVIEW

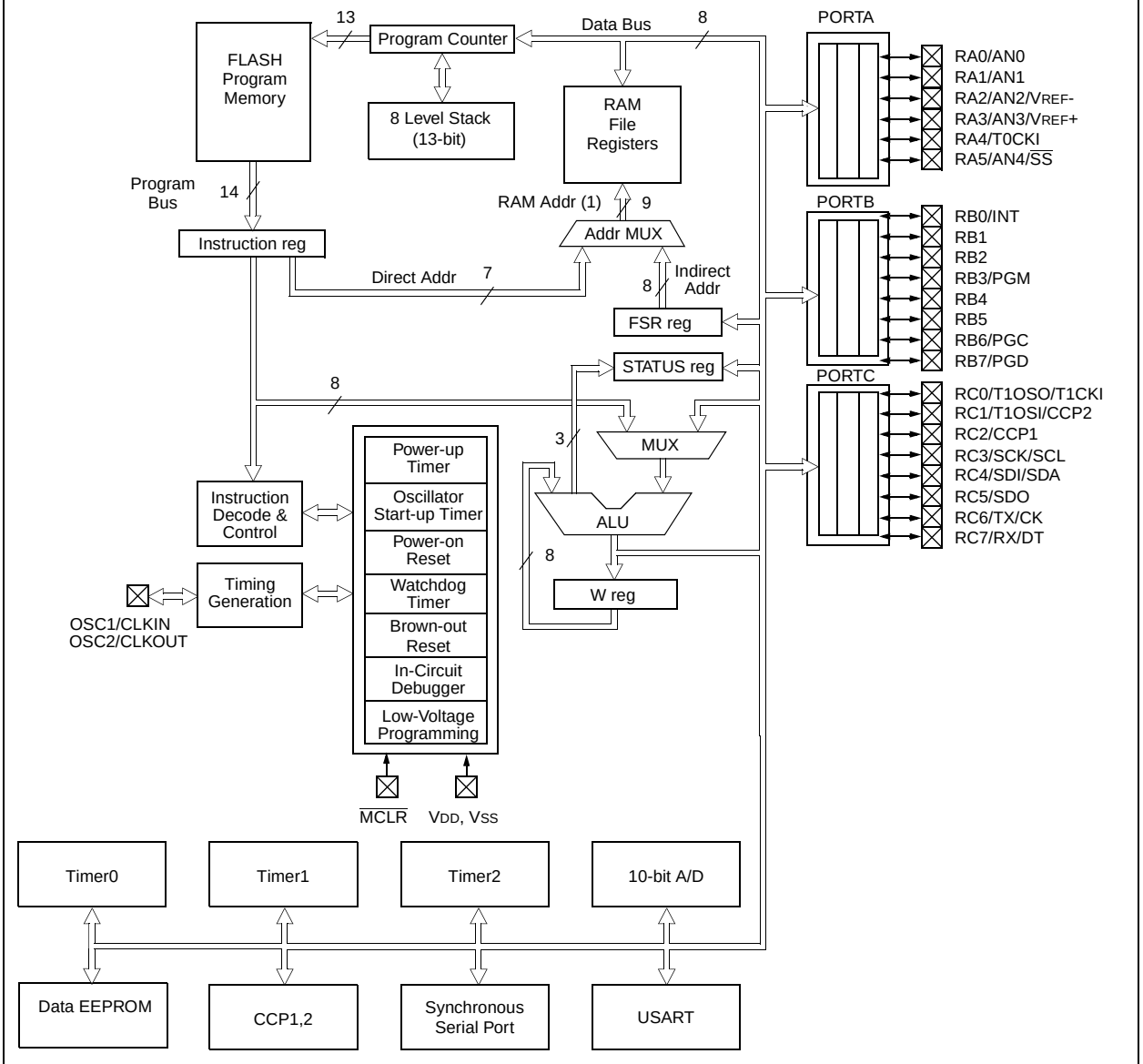
This document contains device-specific information. Additional information may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip website. The Reference Manual should be considered a complementary document to this data sheet, and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

There are four devices (PIC16F873, PIC16F874, PIC16F876 and PIC16F877) covered by this data sheet. The PIC16F876/873 devices come in 28-pin packages and the PIC16F877/874 devices come in 40-pin packages. The 28-pin devices do not have a Parallel Slave Port implemented.

The following two figures are device block diagrams sorted by pin number; 28-pin for Figure 1-1 and 40-pin for Figure 1-2. The 28-pin and 40-pin pinouts are listed in Table 1-1 and Table 1-2, respectively.

FIGURE 1-1: PIC16F873 AND PIC16F876 BLOCK DIAGRAM

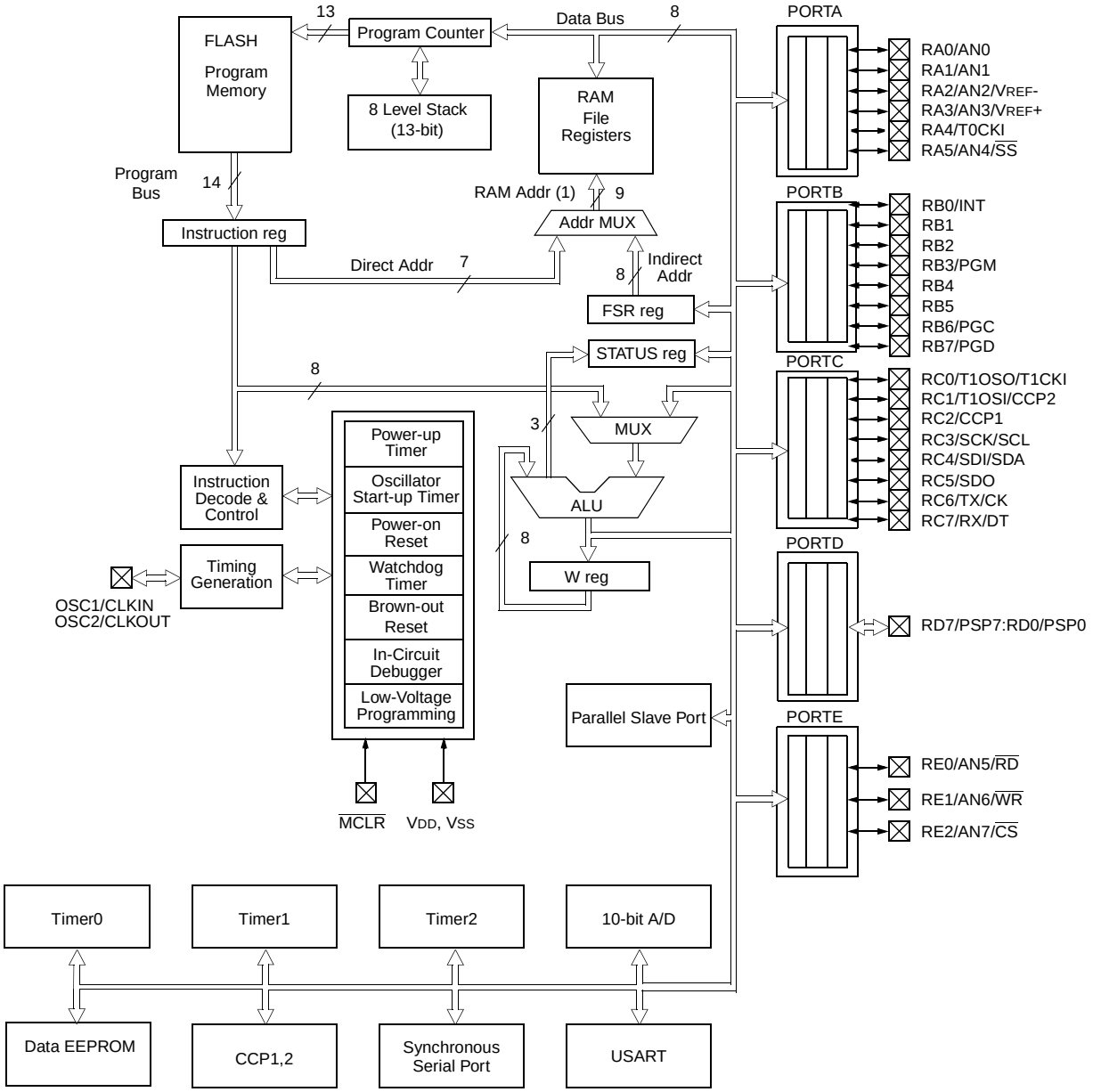
Device	Program FLASH	Data Memory	Data EEPROM
PIC16F873	4K	192 Bytes	128 Bytes
PIC16F876	8K	368 Bytes	256 Bytes



Note 1: Higher order bits are from the STATUS register.

FIGURE 1-2: PIC16F874 AND PIC16F877 BLOCK DIAGRAM

Device	Program FLASH	Data Memory	Data EEPROM
PIC16F874	4K	192 Bytes	128 Bytes
PIC16F877	8K	368 Bytes	256 Bytes



Note 1: Higher order bits are from the STATUS register.

© 1999 Microchip Technology Inc.

TABLE 1-1: PIC16F873 AND PIC16F876 PINOUT DESCRIPTION

Pin Name	DIP Pin#	SOIC Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	9	9	I	ST/CMOS(3)	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	10	10	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, the OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
$\overline{MCLR}$/VPP/THV	1	1	I/P	ST	Master clear (reset) input or programming voltage input or high voltage test mode control. This pin is an active low reset to the device.
					PORTA is a bi-directional I/O port.
RA0/AN0	2	2	I/O	TTL	RA0 can also be analog input0
RA1/AN1	3	3	I/O	TTL	RA1 can also be analog input1
RA2/AN2/VREF-	4	4	I/O	TTL	RA2 can also be analog input2 or negative analog reference voltage
RA3/AN3/VREF+	5	5	I/O	TTL	RA3 can also be analog input3 or positive analog reference voltage
RA4/T0CKI	6	6	I/O	ST	RA4 can also be the clock input to the Timer0 module. Output is open drain type.
RA5/$\overline{SS}$/AN4	7	7	I/O	TTL	RA5 can also be analog input4 or the slave select for the synchronous serial port.
					PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT	21	21	I/O	TTL/ST(1)	RB0 can also be the external interrupt pin.
RB1	22	22	I/O	TTL	
RB2	23	23	I/O	TTL	
RB3/PGM	24	24	I/O	TTL	RB3 can also be the low voltage programming input
RB4	25	25	I/O	TTL	Interrupt on change pin.
RB5	26	26	I/O	TTL	Interrupt on change pin.
RB6/PGC	27	27	I/O	TTL/ST(2)	Interrupt on change pin or In-Circuit Debugger pin. Serial programming clock.
RB7/PGD	28	28	I/O	TTL/ST(2)	Interrupt on change pin or In-Circuit Debugger pin. Serial programming data.
					PORTC is a bi-directional I/O port.
RC0/T1OSO/T1CKI	11	11	I/O	ST	RC0 can also be the Timer1 oscillator output or Timer1 clock input.
RC1/T1OSI/CCP2	12	12	I/O	ST	RC1 can also be the Timer1 oscillator input or Capture2 input/ Compare2 output/PWM2 output.
RC2/CCP1	13	13	I/O	ST	RC2 can also be the Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	14	14	I/O	ST	RC3 can also be the synchronous serial clock input/output for both SPI and I^2C modes.
RC4/SDI/SDA	15	15	I/O	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I^2C mode).
RC5/SDO	16	16	I/O	ST	RC5 can also be the SPI Data Out (SPI mode).
RC6/TX/CK	17	17	I/O	ST	RC6 can also be the USART Asynchronous Transmit or Synchronous Clock.
RC7/RX/DT	18	18	I/O	ST	RC7 can also be the USART Asynchronous Receive or Synchronous Data.
VSS	8, 19	8, 19	P	—	Ground reference for logic and I/O pins.
VDD	20	20	P	—	Positive supply for logic and I/O pins.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in serial programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

TABLE 1-2: PIC16F874 AND PIC16F877 PINOUT DESCRIPTION

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	13	14	30	I	ST/CMOS(4)	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	14	15	31	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
$\overline{MCLR}$/VPP/THV	1	2	18	I/P	ST	Master clear (reset) input or programming voltage input or high voltage test mode control. This pin is an active low reset to the device.
						PORTA is a bi-directional I/O port.
RA0/AN0	2	3	19	I/O	TTL	RA0 can also be analog input0
RA1/AN1	3	4	20	I/O	TTL	RA1 can also be analog input1
RA2/AN2/VREF-	4	5	21	I/O	TTL	RA2 can also be analog input2 or negative analog reference voltage
RA3/AN3/VREF+	5	6	22	I/O	TTL	RA3 can also be analog input3 or positive analog reference voltage
RA4/T0CKI	6	7	23	I/O	ST	RA4 can also be the clock input to the Timer0 timer/counter. Output is open drain type.
RA5/$\overline{SS}$/AN4	7	8	24	I/O	TTL	RA5 can also be analog input4 or the slave select for the synchronous serial port.
						PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT	33	36	8	I/O	TTL/ST(1)	RB0 can also be the external interrupt pin.
RB1	34	37	9	I/O	TTL	
RB2	35	38	10	I/O	TTL	
RB3/PGM	36	39	11	I/O	TTL	RB3 can also be the low voltage programming input
RB4	37	41	14	I/O	TTL	Interrupt on change pin.
RB5	38	42	15	I/O	TTL	Interrupt on change pin.
RB6/PGC	39	43	16	I/O	TTL/ST(2)	Interrupt on change pin or In-Circuit Debugger pin. Serial programming clock.
RB7/PGD	40	44	17	I/O	TTL/ST(2)	Interrupt on change pin or In-Circuit Debugger pin. Serial programming data.
						PORTC is a bi-directional I/O port.
RC0/T1OSO/T1CKI	15	16	32	I/O	ST	RC0 can also be the Timer1 oscillator output or a Timer1 clock input.
RC1/T1OSI/CCP2	16	18	35	I/O	ST	RC1 can also be the Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output.
RC2/CCP1	17	19	36	I/O	ST	RC2 can also be the Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	18	20	37	I/O	ST	RC3 can also be the synchronous serial clock input/output for both SPI and I^2C modes.
RC4/SDI/SDA	23	25	42	I/O	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I^2C mode).
RC5/SDO	24	26	43	I/O	ST	RC5 can also be the SPI Data Out (SPI mode).
RC6/TX/CK	25	27	44	I/O	ST	RC6 can also be the USART Asynchronous Transmit or Synchronous Clock.
RC7/RX/DT	26	29	1	I/O	ST	RC7 can also be the USART Asynchronous Receive or Synchronous Data.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as an external interrupt.
2: This buffer is a Schmitt Trigger input when used in serial programming mode.
3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

© 1999 Microchip Technology Inc.

TABLE 1-2: PIC16F874 AND PIC16F877 PINOUT DESCRIPTION (CONTINUED)

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
						PORTD is a bi-directional I/O port or parallel slave port when interfacing to a microprocessor bus.
RD0/PSP0	19	21	38	I/O	ST/TTL(3)	
RD1/PSP1	20	22	39	I/O	ST/TTL(3)	
RD2/PSP2	21	23	40	I/O	ST/TTL(3)	
RD3/PSP3	22	24	41	I/O	ST/TTL(3)	
RD4/PSP4	27	30	2	I/O	ST/TTL(3)	
RD5/PSP5	28	31	3	I/O	ST/TTL(3)	
RD6/PSP6	29	32	4	I/O	ST/TTL(3)	
RD7/PSP7	30	33	5	I/O	ST/TTL(3)	
						PORTE is a bi-directional I/O port.
RE0/$\overline{RD}$/AN5	8	9	25	I/O	ST/TTL(3)	RE0 can also be read control for the parallel slave port, or analog input5.
RE1/$\overline{WR}$/AN6	9	10	26	I/O	ST/TTL(3)	RE1 can also be write control for the parallel slave port, or analog input6.
RE2/$\overline{CS}$/AN7	10	11	27	I/O	ST/TTL(3)	RE2 can also be select control for the parallel slave port, or analog input7.
VSS	12,31	13,34	6,29	P	—	Ground reference for logic and I/O pins.
VDD	11,32	12,35	7,28	P	—	Positive supply for logic and I/O pins.
NC	—	1,17,28, 40	12,13, 33,34		—	These pins are not internally connected. These pins should be left unconnected.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as an external interrupt.
2: This buffer is a Schmitt Trigger input when used in serial programming mode.
3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

NOTES:

© 1999 Microchip Technology Inc.

2.0 MEMORY ORGANIZATION

There are three memory blocks in each of these PICmicro MCUs. The Program Memory and Data Memory have separate buses so that concurrent access can occur and is detailed in this section. The EEPROM data memory block is detailed in Section 4.0.

Additional information on device memory may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

2.1 Program Memory Organization

The PIC16F87X devices have a 13-bit program counter capable of addressing an 8K x 14 program memory space. The PIC16F877/876 devices have 8K x 14 words of FLASH program memory and the PIC16F873/874 devices have 4K x 14. Accessing a location above the physically implemented address will cause a wrap-around.

The reset vector is at 0000h and the interrupt vector is at 0004h.

FIGURE 2-1: PIC16F877/876 PROGRAM MEMORY MAP AND STACK

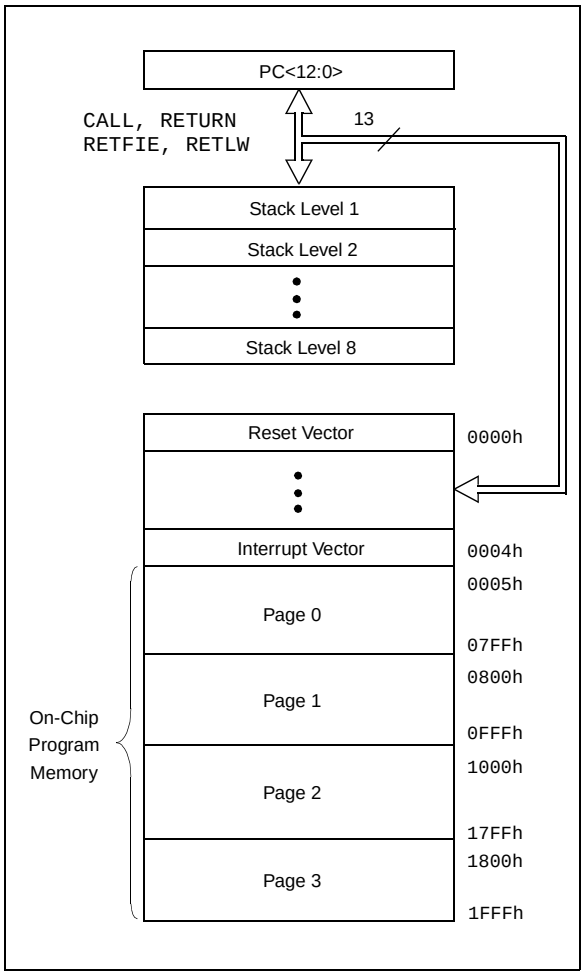
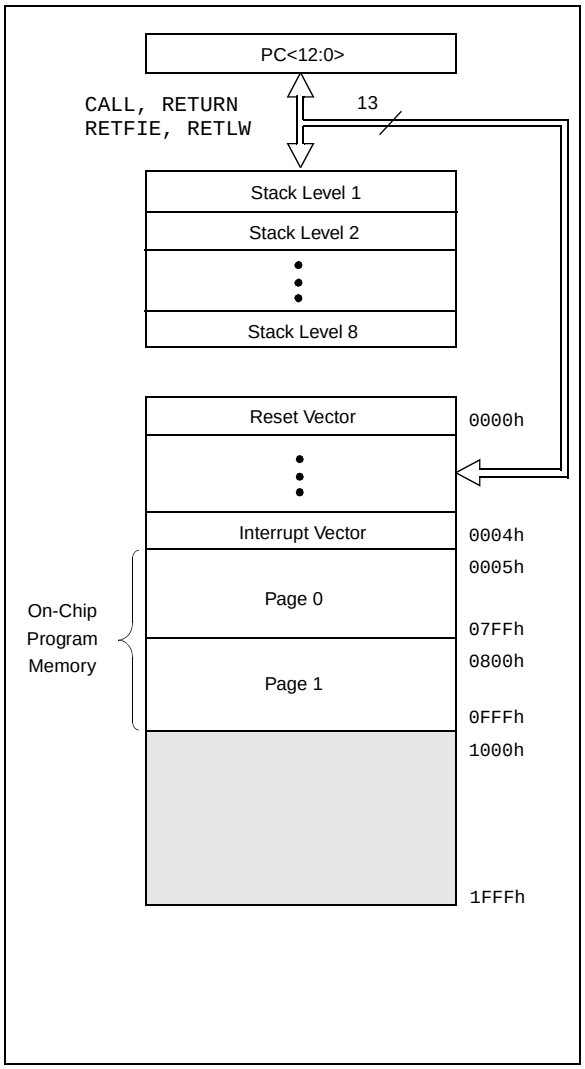


FIGURE 2-2: PIC16F874/873 PROGRAM MEMORY MAP AND STACK



2.2 Data Memory Organization

The data memory is partitioned into multiple banks which contain the General Purpose Registers and the Special Function Registers. Bits RP1(STATUS<6>) and RP0 (STATUS<5>) are the bank select bits.

RP1:RP0	Bank
00	0
01	1
10	2
11	3

Each bank extends up to 7Fh (128 bytes). The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers, implemented as static RAM. All implemented banks contain Special Function Registers. Some “high use” Special Function Registers from one bank may be mirrored in another bank for code reduction and quicker access.

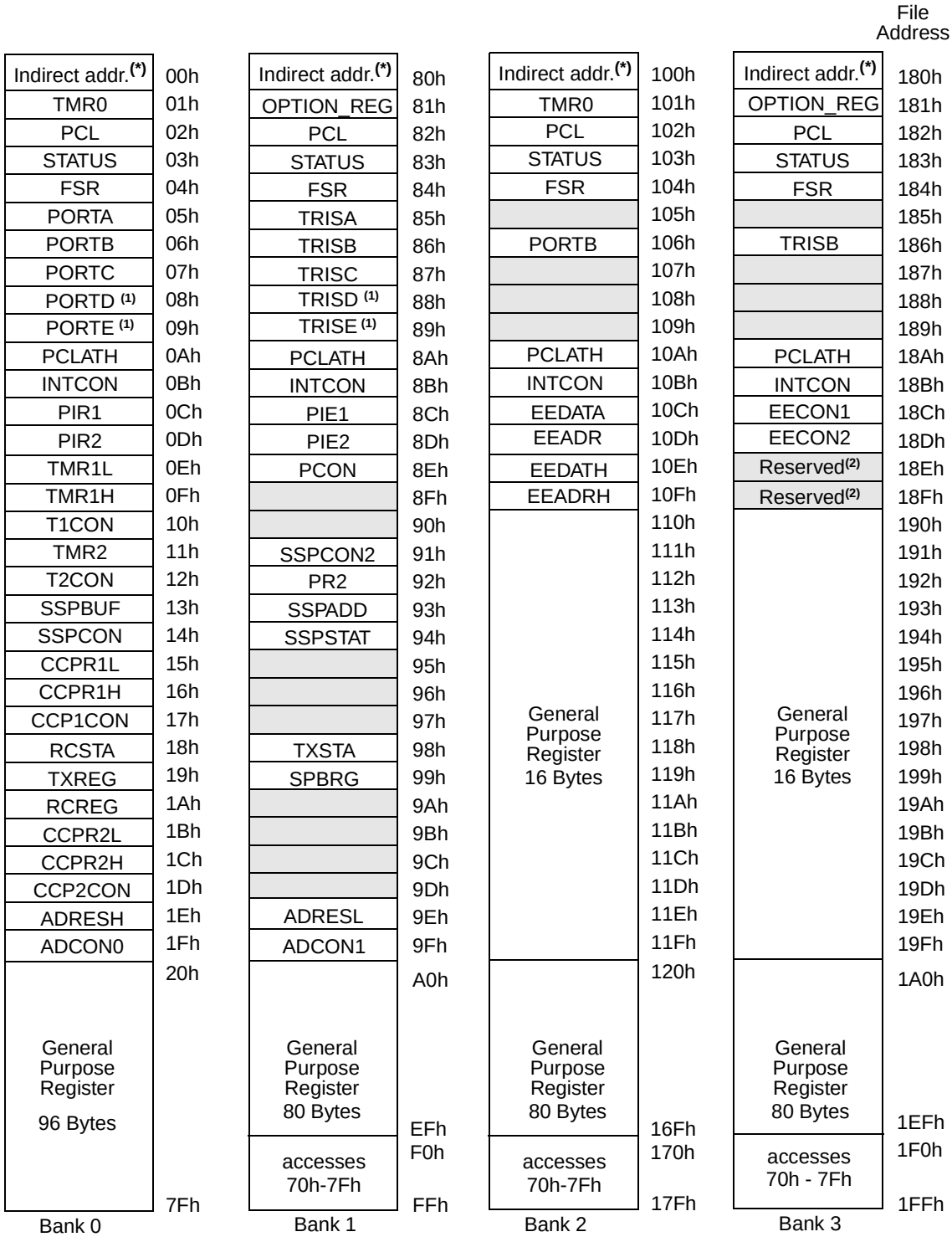
Note:	EEPROM Data Memory description can be found in Section 4.0 of this Data Sheet

2.2.1 GENERAL PURPOSE REGISTER FILE

The register file can be accessed either directly, or indirectly through the File Select Register FSR.

© 1999 Microchip Technology Inc.

FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP



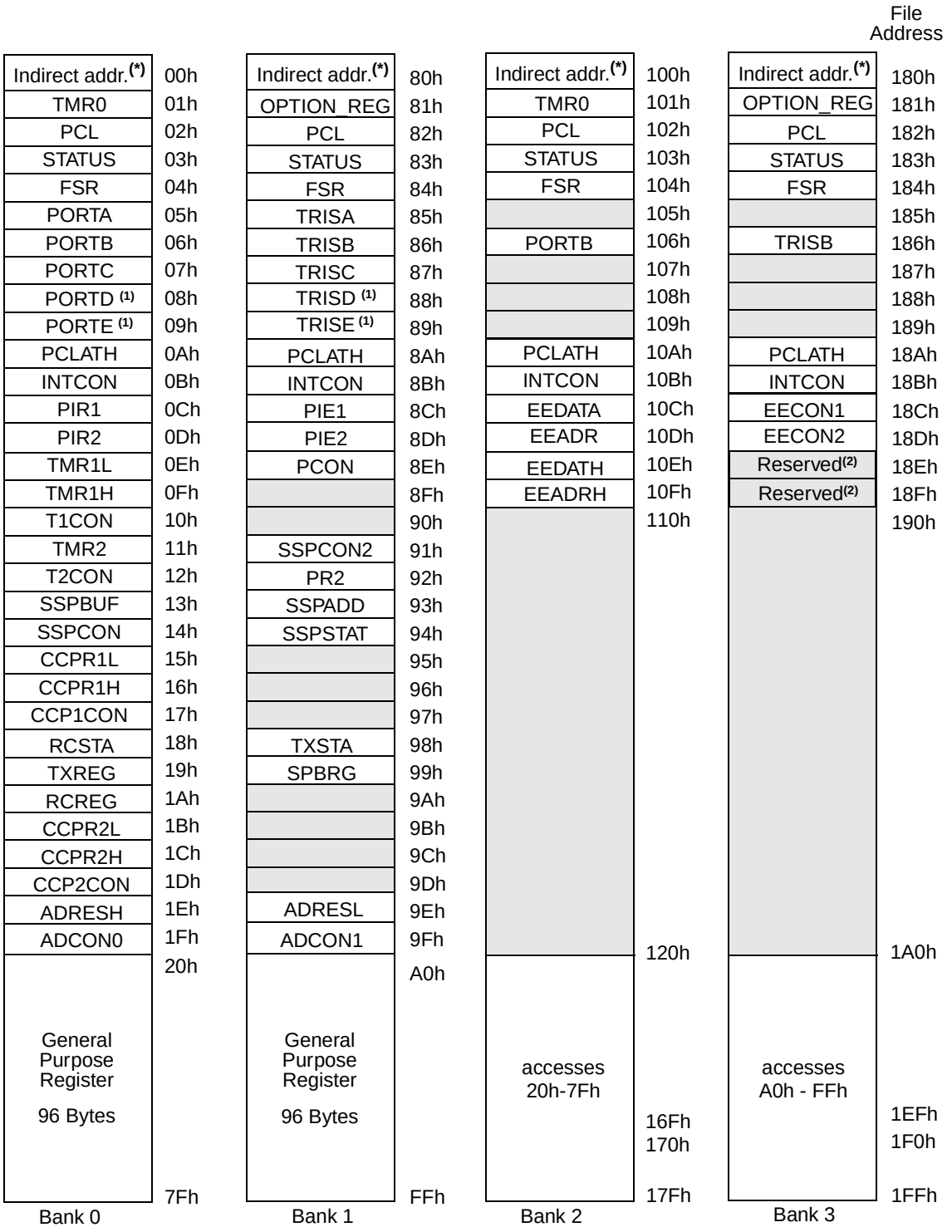
Unimplemented data memory locations, read as '0'.

* Not a physical register.

Note 1: These registers are not implemented on 28-pin devices.

2: These registers are reserved, maintain these registers clear.

FIGURE 2-4: PIC16F874/873 REGISTER FILE MAP



Unimplemented data memory locations, read as '0'.

* Not a physical register.

Note 1: These registers are not implemented on 28-pin devices.

2: These registers are reserved, maintain these registers clear.

© 1999 Microchip Technology Inc.

2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral modules for controlling the desired operation of the device. These registers are implemented as static RAM. A list of these registers is given in Table 2-1.

The Special Function Registers can be classified into two sets; core (CPU) and peripheral. Those registers associated with the core functions are described in detail in this section. Those related to the operation of the peripheral features are described in detail in the peripheral feature section.

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (2)
Bank 0											
00h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h(4)	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--0x 0000	--0u 0000
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
07h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	uuuu uuuu
08h(5)	PORTD	PORTD Data Latch when written: PORTD pins when read								xxxx xxxx	uuuu uuuu
09h(5)	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
0Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(3)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh	PIR2	—	(6)	—	EEIF	BCLIF	—	—	CCP2IF	-r-0 0--0	-r-0 0--0
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	--uu uuuu
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
19h	TXREG	USART Transmit Data Register								0000 0000	0000 0000
1Ah	RCREG	USART Receive Data Register								0000 0000	0000 0000
1Bh	CCPR2L	Capture/Compare/PWM Register2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM Register2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
1Eh	ADRESH	A/D Result Register High Byte								xxxx xxxx	uuuu uuuu
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	—	ADON	0000 00-0	0000 00-0

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through $\overline{MCLR}$ and Watchdog Timer Reset.

3: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

4: These registers can be addressed from any bank.

5: PORTD, PORTE, TRISD, and TRISE are not physically implemented on the 28-pin devices, read as '0'.

6: PIR2<6> and PIE2<6> are reserved on these devices; always maintain these bits clear.

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (2)
Bank 1											
80h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION_REG	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h(4)	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
84h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
88h(5)	TRISD	PORTD Data Direction Register								1111 1111	1111 1111
89h(5)	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111
8Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	PSPIE(3)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh	PIE2	—	(6)	—	EEIE	BCLIE	—	—	CCP2IE	-r-0 0--0	-r-0 0--0
8Eh	PCON	—	—	—	—	—	—	$\overline{POR}$	BOR	---- --qq	---- --uu
8Fh	—	Unimplemented								—	—
90h	—	Unimplemented								—	—
91h	SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
92h	PR2	Timer2 Period Register								1111 1111	1111 1111
93h	SSPADD	Synchronous Serial Port (I^2C mode) Address Register								0000 0000	0000 0000
94h	SSPSTAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	0000 0000
95h	—	Unimplemented								—	—
96h	—	Unimplemented								—	—
97h	—	Unimplemented								—	—
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000
9Ah	—	Unimplemented								—	—
9Bh	—	Unimplemented								—	—
9Ch	—	Unimplemented								—	—
9Dh	—	Unimplemented								—	—
9Eh	ADRESL	A/D Result Register Low Byte								xxxx xxxx	uuuu uuuu
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	0--- 0000	0--- 0000

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through $\overline{MCLR}$ and Watchdog Timer Reset.

3: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

4: These registers can be addressed from any bank.

5: PORTD, PORTE, TRISD, and TRISE are not physically implemented on the 28-pin devices, read as '0'.

6: PIR2<6> and PIE2<6> are reserved on these devices; always maintain these bits clear.

© 1999 Microchip Technology Inc.

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (2)
Bank 2											
100h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
101h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
102h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
103h(4)	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxx	000q quuu
104h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
105h	—	Unimplemented								—	—
106h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
107h	—	Unimplemented								—	—
108h	—	Unimplemented								—	—
109h	—	Unimplemented								—	—
10Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
10Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
10Ch	EEDATA	EEPROM data register								xxxx xxxx	uuuu uuuu
10Dh	EEADR	EEPROM address register								xxxx xxxx	uuuu uuuu
10Eh	EEDATH	—	—	EEPROM data register high byte						xxxx xxxx	uuuu uuuu
10Fh	EEADRH	—	—	—	EEPROM address register high byte					xxxx xxxx	uuuu uuuu
Bank 3											
180h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
181h	OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
182h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
183h(4)	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxx	000q quuu
184h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
185h	—	Unimplemented								—	—
186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
187h	—	Unimplemented								—	—
188h	—	Unimplemented								—	—
189h	—	Unimplemented								—	—
18Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
18Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
18Ch	EECON1	EEPGD	—	—	—	WRERR	WREN	WR	RD	x--- x000	x--- u000
18Dh	EECON2	EEPROM control register2 (not a physical register)								---- ----	---- ----
18Eh	—	Reserved maintain clear								0000 0000	0000 0000
18Fh	—	Reserved maintain clear								0000 0000	0000 0000

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through $\overline{\text{MCLR}}$ and Watchdog Timer Reset.

3: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

4: These registers can be addressed from any bank.

5: PORTD, PORTE, TRISD, and TRISE are not physically implemented on the 28-pin devices, read as '0'.

6: PIR2<6> and PIE2<6> are reserved on these devices; always maintain these bits clear.

2.2.2.1 STATUS REGISTER

The STATUS register contains the arithmetic status of the ALU, the RESET status and the bank select bits for data memory.

The STATUS register can be the destination for any instruction, as with any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{TO}$ and $\overline{PD}$ bits are not writable, therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, CLRF STATUS will clear the upper-three bits and set the Z bit. This leaves the STATUS register as 000u u1uu (where u = unchanged).

It is recommended, therefore, that only BCF, BSF, SWAPF and MOVWF instructions are used to alter the STATUS register, because these instructions do not affect the Z, C or DC bits from the STATUS register. For other instructions not affecting any status bits, see the "Instruction Set Summary."

Note 1: The C and DC bits operate as a $\overline{borrow}$ and $\overline{digit\ borrow}$ bit, respectively, in subtraction. See the SUBLW and SUBWF instructions for examples.

REGISTER 2-1: STATUS REGISTER (ADDRESS 03h, 83h, 103h, 183h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n= Value at POR reset

bit 7: **IRP:** Register Bank Select bit (used for indirect addressing)
1 = Bank 2, 3 (100h - 1FFh)
0 = Bank 0, 1 (00h - FFh)

bit 6-5: **RP1:RP0**: Register Bank Select bits (used for direct addressing)
11 = Bank 3 (180h - 1FFh)
10 = Bank 2 (100h - 17Fh)
01 = Bank 1 (80h - FFh)
00 = Bank 0 (00h - 7Fh)
Each bank is 128 bytes

bit 4: **$\overline{TO}$**: Time-out bit
1 = After power-up, CLRWDT instruction, or SLEEP instruction
0 = A WDT time-out occurred

bit 3: **$\overline{PD}$**: Power-down bit
1 = After power-up or by the CLRWDT instruction
0 = By execution of the SLEEP instruction

bit 2: **Z**: Zero bit
1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

bit 1: **DC**: Digit carry/$\overline{borrow}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)
(for $\overline{borrow}$ the polarity is reversed)
1 = A carry-out from the 4th low order bit of the result occurred
0 = No carry-out from the 4th low order bit of the result

bit 0: **C**: Carry/$\overline{borrow}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)
1 = A carry-out from the most significant bit of the result occurred
0 = No carry-out from the most significant bit of the result occurred
Note: For $\overline{borrow}$ the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (RRF, RLF) instructions, this bit is loaded with either the high or low order bit of the source register.

© 1999 Microchip Technology Inc.

2.2.2.2 OPTION_REG REGISTER

The OPTION_REG Register is a readable and writable register, which contains various control bits to configure the TMR0 prescaler/WDT postscaler (single assignable register known also as the prescaler), the External INT Interrupt, TMR0 and the weak pull-ups on PORTB.

Note: To achieve a 1:1 prescaler assignment for the TMR0 register, assign the prescaler to the Watchdog Timer.

REGISTER 2-2: OPTION_REG REGISTER (ADDRESS 81h, 181h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n= Value at POR reset

bit 7: **$\overline{\text{RBPU}}$: PORTB Pull-up Enable bit**
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual port latch values

bit 6: **INTEDG**: Interrupt Edge Select bit
1 = Interrupt on rising edge of RB0/INT pin
0 = Interrupt on falling edge of RB0/INT pin

bit 5: **T0CS**: TMR0 Clock Source Select bit
1 = Transition on RA4/T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4: **T0SE**: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on RA4/T0CKI pin
0 = Increment on low-to-high transition on RA4/T0CKI pin

bit 3: **PSA**: Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0: **PS2:PS0**: Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

Note: When using low voltage ICSP programming (LVP) and the pull-ups on PORTB are enabled, bit 3 in the TRISB register must be cleared to disable the pull-up on RB3 and ensure the proper operation of the device.

2.2.2.3 INTCON REGISTER

The INTCON Register is a readable and writable register, which contains various enable and flag bits for the TMR0 register overflow, RB Port change and External RB0/INT pin interrupts.

Note:	Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-3: INTCON REGISTER (ADDRESS 0Bh, 8Bh, 10Bh, 18Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n= Value at POR reset

bit 7: **GIE:** Global Interrupt Enable bit
1 = Enables all un-masked interrupts
0 = Disables all interrupts

bit 6: **PEIE**: Peripheral Interrupt Enable bit
1 = Enables all un-masked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5: **T0IE**: TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4: **INTE**: RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3: **RBIE**: RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2: **T0IF**: TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1: **INTF**: RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (must be cleared in software)
0 = The RB0/INT external interrupt did not occur

bit 0: **RBIF**: RB Port Change Interrupt Flag bit
1 = At least one of the RB7:RB4 pins changed state (must be cleared in software)
0 = None of the RB7:RB4 pins have changed state

© 1999 Microchip Technology Inc.

2.2.2.4 PIE1 REGISTER

The PIE1 register contains the individual enable bits for the peripheral interrupts.

Note:	Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

REGISTER 2-4: PIE1 REGISTER (ADDRESS 8Ch)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n= Value at POR reset

bit 7: **PSPIE(1):** Parallel Slave Port Read/Write Interrupt Enable bit
1 = Enables the PSP read/write interrupt
0 = Disables the PSP read/write interrupt

bit 6: **ADIE**: A/D Converter Interrupt Enable bit
1 = Enables the A/D converter interrupt
0 = Disables the A/D converter interrupt

bit 5: **RCIE**: USART Receive Interrupt Enable bit
1 = Enables the USART receive interrupt
0 = Disables the USART receive interrupt

bit 4: **TXIE**: USART Transmit Interrupt Enable bit
1 = Enables the USART transmit interrupt
0 = Disables the USART transmit interrupt

bit 3: **SSPIE**: Synchronous Serial Port Interrupt Enable bit
1 = Enables the SSP interrupt
0 = Disables the SSP interrupt

bit 2: **CCP1IE**: CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt

bit 1: **TMR2IE**: TMR2 to PR2 Match Interrupt Enable bit
1 = Enables the TMR2 to PR2 match interrupt
0 = Disables the TMR2 to PR2 match interrupt

bit 0: **TMR1IE**: TMR1 Overflow Interrupt Enable bit
1 = Enables the TMR1 overflow interrupt
0 = Disables the TMR1 overflow interrupt

Note 1: PSPIE is reserved on 28-pin devices; always maintain this bit clear.

2.2.2.5 PIR1 REGISTER

The PIR1 register contains the individual flag bits for the peripheral interrupts.

Note: Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt bits are clear prior to enabling an interrupt.

REGISTER 2-5: PIR1 REGISTER (ADDRESS 0Ch)

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit7							bit0

R = Readable bit
W = Writable bit
- n= Value at POR reset

bit 7: **PSPIF(1):** Parallel Slave Port Read/Write Interrupt Flag bit
1 = A read or a write operation has taken place (must be cleared in software)
0 = No read or write has occurred

bit 6: **ADIF**: A/D Converter Interrupt Flag bit
1 = An A/D conversion completed
0 = The A/D conversion is not complete

bit 5: **RCIF**: USART Receive Interrupt Flag bit
1 = The USART receive buffer is full
0 = The USART receive buffer is empty

bit 4: **TXIF**: USART Transmit Interrupt Flag bit
1 = The USART transmit buffer is empty
0 = The USART transmit buffer is full

bit 7: **SSPIF**: Synchronous Serial Port (SSP) Interrupt Flag
1 = The SSP interrupt condition has occurred, and must be cleared in software before returning from the interrupt service routine. The conditions that will set this bit are:
SPI
A transmission/reception has taken place.
I^2C Slave
A transmission/reception has taken place.
I^2C Master
A transmission/reception has taken place.
The initiated start condition was completed by the SSP module.
The initiated stop condition was completed by the SSP module.
The initiated restart condition was completed by the SSP module.
The initiated acknowledge condition was completed by the SSP module.
A start condition occurred while the SSP module was idle (Multimaster system).
A stop condition occurred while the SSP module was idle (Multimaster system).
0 = No SSP interrupt condition has occurred.

bit 2: **CCP1IF**: CCP1 Interrupt Flag bit
Capture Mode
1 = A TMR1 register capture occurred (must be cleared in software)
0 = No TMR1 register capture occurred
Compare Mode
1 = A TMR1 register compare match occurred (must be cleared in software)
0 = No TMR1 register compare match occurred
PWM Mode
Unused in this mode

bit 1: **TMR2IF**: TMR2 to PR2 Match Interrupt Flag bit
1 = TMR2 to PR2 match occurred (must be cleared in software)
0 = No TMR2 to PR2 match occurred

bit 0: **TMR1IF**: TMR1 Overflow Interrupt Flag bit
1 = TMR1 register overflowed (must be cleared in software)
0 = TMR1 register did not overflow

Note 1: PSPIF is reserved on 28-pin devices; always maintain this bit clear.

© 1999 Microchip Technology Inc.

2.2.2.6 PIE2 REGISTER

The PIE2 register contains the individual enable bits for the CCP2 peripheral interrupt, the SSP bus collision interrupt, and the EEPROM write operation interrupt.

REGISTER 2-6: PIE2 REGISTER (ADDRESS 8Dh)

U-0	R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0
—	—	—	EEIE	BCLIE	—	—	CCP2IE
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n= Value at POR reset

bit 7: **Unimplemented:** Read as '0'

bit 6: **Reserved**: Always maintain this bit clear

bit 5: **Unimplemented:** Read as '0'

bit 4: **EEIE**: EEPROM Write Operation Interrupt Enable
1 = Enable EE Write Interrupt
0 = Disable EE Write Interrupt

bit 3: **BCLIE**: Bus Collision Interrupt Enable
1 = Enable Bus Collision Interrupt
0 = Disable Bus Collision Interrupt

bit 2-1: **Unimplemented:** Read as '0'

bit 0: **CCP2IE**: CCP2 Interrupt Enable bit
1 = Enables the CCP2 interrupt
0 = Disables the CCP2 interrupt

2.2.2.7 PIR2 REGISTER

The PIR2 register contains the flag bits for the CCP2 interrupt, the SSP bus collision interrupt and the EEPROM write operation interrupt.

Note:	Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-7: PIR2 REGISTER (ADDRESS 0Dh)

U-0	R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0
—	—	—	EEIF	BCLIF	—	—	CCP2IF
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n= Value at POR reset

bit 7: **Unimplemented:** Read as '0'

bit 6: **Reserved:** Always maintain this bit clear

bit 5: **Unimplemented:** Read as '0'

bit 4: **EEIF**: EEPROM Write Operation Interrupt Flag bit
1 = The write operation completed (must be cleared in software)
0 = The write operation is not complete or has not been started

bit 3: **BCLIF**: Bus Collision Interrupt Flag
1 = A bus collision has occurred in the SSP, when configured for I^2C master mode
0 = No bus collision has occurred

bit 2-1: **Unimplemented:** Read as '0'

bit 0: **CCP2IF**: CCP2 Interrupt Flag bit
Capture Mode
1 = A TMR1 register capture occurred (must be cleared in software)
0 = No TMR1 register capture occurred
Compare Mode
1 = A TMR1 register compare match occurred (must be cleared in software)
0 = No TMR1 register compare match occurred
PWM Mode
Unused

© 1999 Microchip Technology Inc.

2.2.2.8 PCON REGISTER

The Power Control (PCON) Register contains flag bits to allow differentiation between a Power-on Reset (POR), a Brown-out Reset (BOR), a Watch-dog Reset (WDT) and an external $\overline{MCLR}$ Reset.

Note:	$\overline{BOR}$ is unknown on POR. It must be set by the user and checked on subsequent rests to see if BOR is clear, indicating a brown-out has occurred. The BOR status bit is a don't care and is not predictable if the brown-out circuit is disabled (by clearing the BODEN bit in the configuration word).

REGISTER 2-8: PCON REGISTER (ADDRESS 8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-1
—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n= Value at POR reset

bit 7-2: **Unimplemented:** Read as '0'

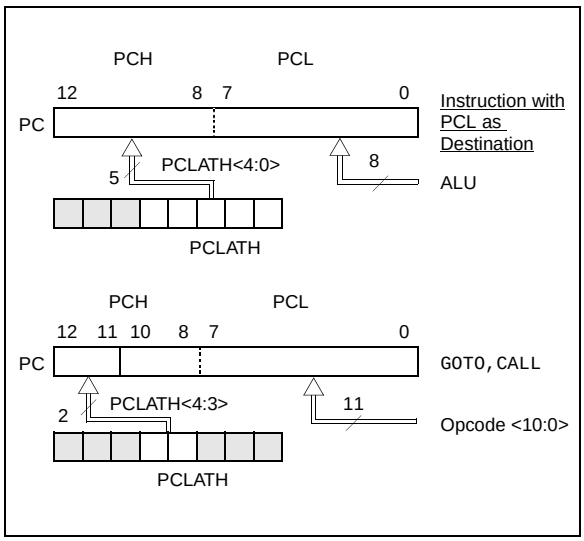
bit 1: **$\overline{POR}$**: Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0: **$\overline{BOR}$**: Brown-out Reset Status bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

2.3 PCL and PCLATH

The program counter (PC) is 13-bits wide. The low byte comes from the PCL register, which is a readable and writable register. The upper bits (PC<12:8>) are not readable, but are indirectly writable through the PCLATH register. On any reset, the upper bits of the PC will be cleared. Figure 2-5 shows the two situations for the loading of the PC. The upper example in the figure shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in the figure shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 2-5: LOADING OF PC IN DIFFERENT SITUATIONS



2.3.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). When doing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256 byte block). Refer to the application note, *"Implementing a Table Read"* (AN556).

2.3.2 STACK

The PIC16CXX family has an 8-level deep x 13-bit wide hardware stack. The stack space is not part of either program or data space and the stack pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

Note 1:	There are no status bits to indicate stack overflow or stack underflow conditions.
2:	There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

2.4 Program Memory Paging

PIC16CXX devices are capable of addressing a continuous 8K word block of program memory. The CALL and GOTO instructions provide only 11 bits of address to allow branching within any 2K program memory page. When doing a CALL or GOTO instruction, the upper 2 bits of the address are provided by PCLATH<4:3>. When doing a CALL or GOTO instruction, the user must ensure that the page select bits are programmed so that the desired program memory page is addressed. If a return from a CALL instruction (or interrupt) is executed, the entire 13-bit PC is popped off the stack. Therefore, manipulation of the PCLATH<4:3> bits are not required for the return instructions (which POPs the address from the stack)

Example 2-1 shows the calling of a subroutine in page 1 of the program memory. This example assumes that PCLATH is saved and restored by the interrupt service routine (if interrupts are used).

EXAMPLE 2-1: CALL OF A SUBROUTINE IN PAGE 1 FROM PAGE 0

```
       ORG  0x500
       BCF  PCLATH,4
       BSF  PCLATH,3   ;Select page 1 (800h-FFFh)
       CALL SUB1_P1    ;Call subroutine in
       :               ;page 1 (800h-FFFh)
       :
       ORG  0x900      ;page 1 (800h-FFFh)
SUB1_P1
       :               ;called subroutine
       :               ;page 1 (800h-FFFh)
       :
       RETURN          ;return to Call subroutine
                       ;in page 0 (000h-7FFh)
```


© 1999 Microchip Technology Inc.

2.5 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

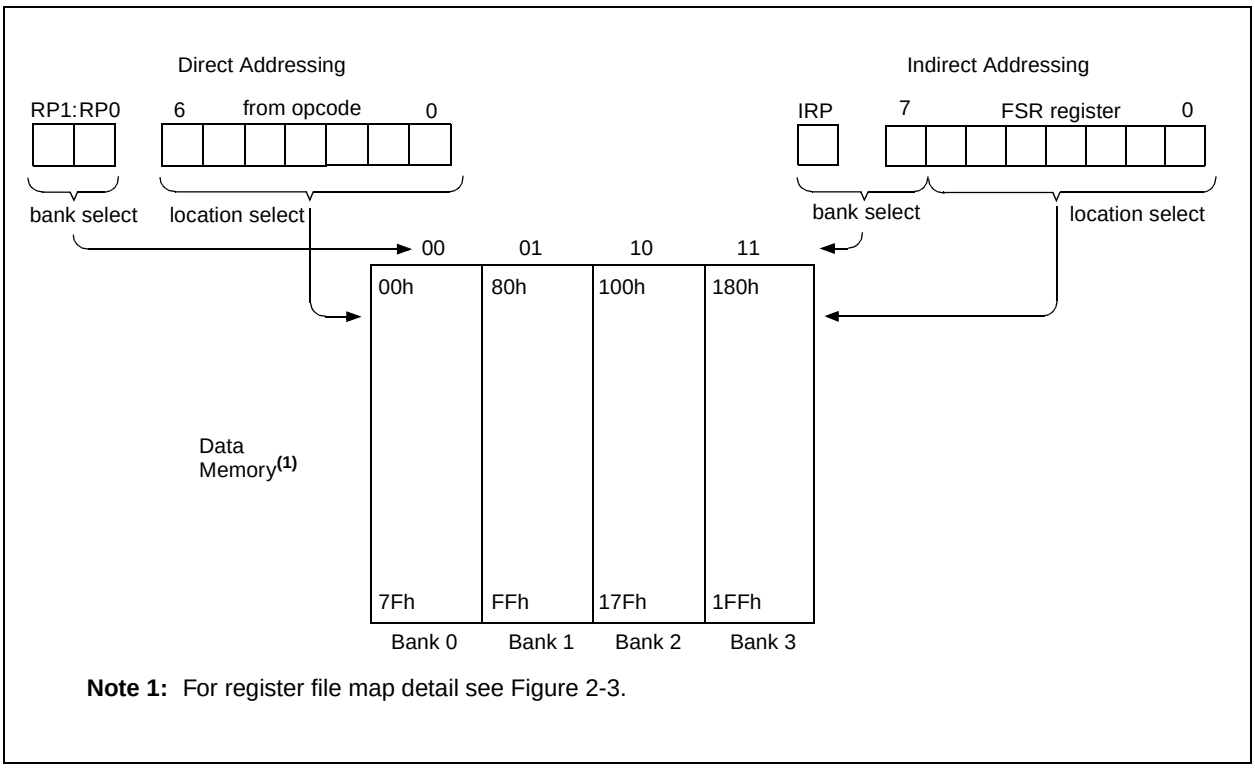
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses the register pointed to by the File Select Register, FSR. Reading the INDF register itself indirectly (FSR = '0') will read 00h. Writing to the INDF register indirectly results in a no-operation (although status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 2-6.

A simple program to clear RAM locations 20h-2Fh using indirect addressing is shown in Example 2-2.

EXAMPLE 2-2: INDIRECT ADDRESSING

```
            movlw   0x20    ;initialize pointer
            movwf   FSR     ;to RAM
NEXT        clrf    INDF    ;clear INDF register
            incf    FSR,F   ;inc pointer
            btfss   FSR,4   ;all done?
            goto    NEXT    ;no clear next
CONTINUE
            :               ;yes continue
```

FIGURE 2-6: DIRECT/INDIRECT ADDRESSING



Note 1: For register file map detail see Figure 2-3.

NOTES:

© 1999 Microchip Technology Inc.

3.0 I/O PORTS

Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Additional information on I/O ports may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

3.1 PORTA and the TRISA Register

PORTA is a 6-bit wide bi-directional port. The corresponding data direction register is TRISA. Setting a TRISA bit (=1) will make the corresponding PORTA pin an input (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISA bit (=0) will make the corresponding PORTA pin an output (i.e., put the contents of the output latch on the selected pin).

Reading the PORTA register reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, the value is modified and then written to the port data latch.

Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin. The RA4/T0CKI pin is a Schmitt Trigger input and an open drain output. All other PORTA pins have TTL input levels and full CMOS output drivers.

Other PORTA pins are multiplexed with analog inputs and analog VREF input. The operation of each pin is selected by clearing/setting the control bits in the ADCON1 register (A/D Control Register1).

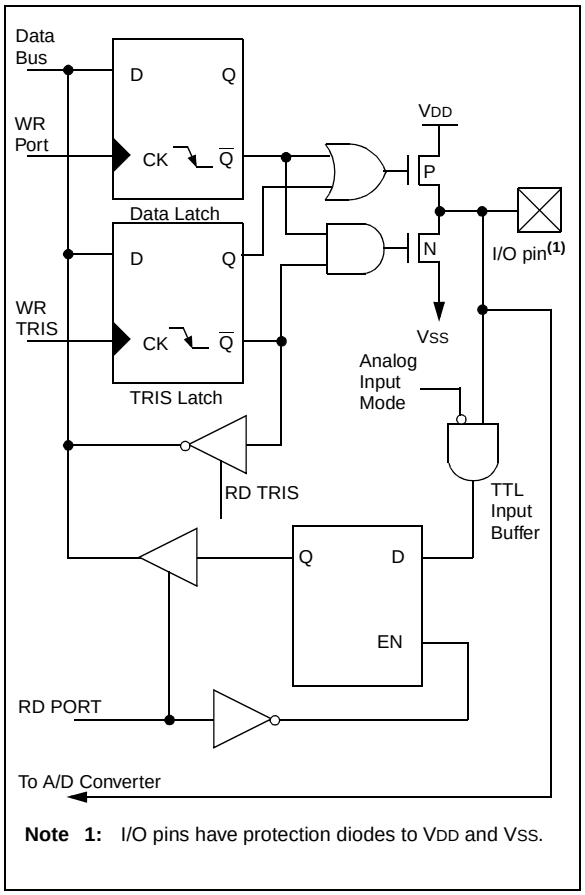
Note:	On a Power-on Reset, these pins are configured as analog inputs and read as '0'.

The TRISA register controls the direction of the RA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

EXAMPLE 3-1: INITIALIZING PORTA

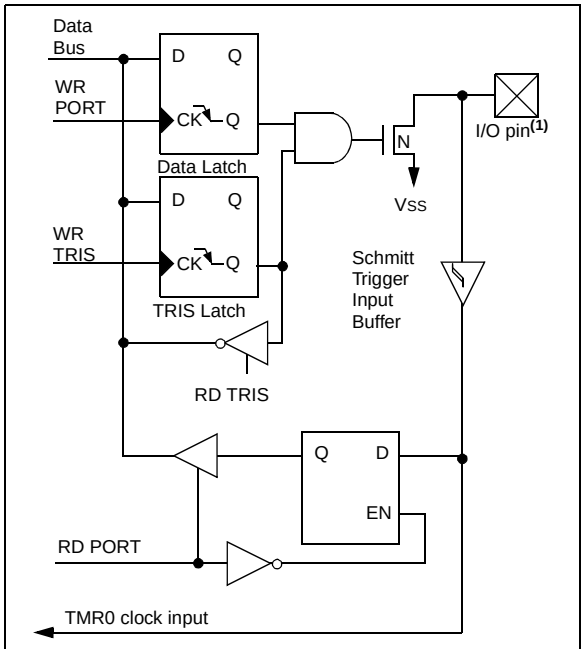
```
BCF     STATUS, RP0     ;
BCF     STATUS, RP1     ; Bank0
CLRF    PORTA           ; Initialize PORTA by
                        ; clearing output
                        ; data latches
BSF     STATUS, RP0     ; Select Bank 1
MOVLW   0x06            ; Configure all pins
MOVWF   ADCON1          ; as digital inputs
MOVLW   0xCF            ; Value used to
                        ; initialize data
                        ; direction
MOVWF   TRISA           ; Set RA<3:0> as inputs
                        ; RA<5:4> as outputs
                        ; TRISA<7:6> are always
                        ; read as '0'.
```

FIGURE 3-1: BLOCK DIAGRAM OF RA3:RA0 AND RA5 PINS



Note 1: I/O pins have protection diodes to VDD and VSS.

FIGURE 3-2: BLOCK DIAGRAM OF RA4/T0CKI PIN



Note 1: I/O pin has protection diodes to VSS only.

TABLE 3-1: PORTA FUNCTIONS

Name	Bit#	Buffer	Function
RA0/AN0	bit0	TTL	Input/output or analog input
RA1/AN1	bit1	TTL	Input/output or analog input
RA2/AN2	bit2	TTL	Input/output or analog input
RA3/AN3/VREF	bit3	TTL	Input/output or analog input or VREF
RA4/T0CKI	bit4	ST	Input/output or external clock input for Timer0 Output is open drain type
RA5/$\overline{SS}$/AN4	bit5	TTL	Input/output or slave select input for synchronous serial port or analog input

Legend: TTL = TTL input, ST = Schmitt Trigger input

TABLE 3-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
05h	PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--0x 0000	--0u 0000
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTA.

Note:	When using the SSP module in SPI slave mode and $\overline{SS}$ enabled, the A/D converter must be set to one of the following modes where PCFG3:PCFG0 = 0100,0101, 011x, 1101, 1110, 1111.

© 1999 Microchip Technology Inc.

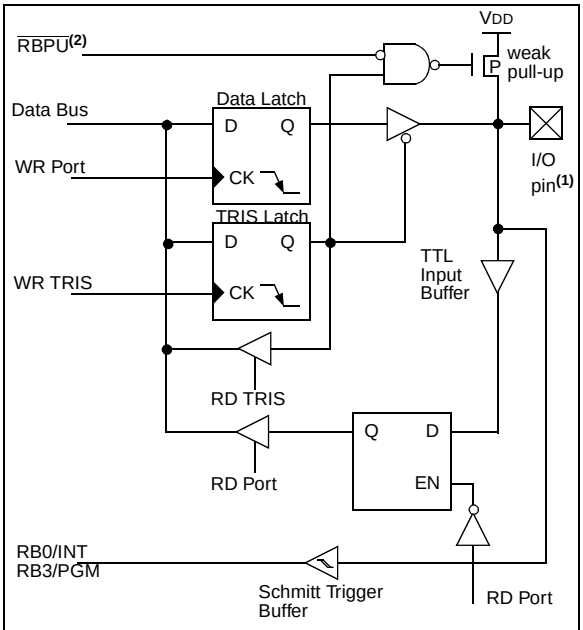
3.2 PORTB and the TRISB Register

PORTB is an 8-bit wide, bi-directional port. The corresponding data direction register is TRISB. Setting a TRISB bit (=1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISB bit (=0) will make the corresponding PORTB pin an output (i.e., put the contents of the output latch on the selected pin).

Three pins of PORTB are multiplexed with the Low Voltage Programming function; RB3/PGM, RB6/PGC and RB7/PGD. The alternate functions of these pins are described in the Special Features Section.

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit $\overline{\text{RBPU}}$ (OPTION_REG<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

FIGURE 3-3: BLOCK DIAGRAM OF RB3:RB0 PINS



Note 1: I/O pins have diode protection to VDD and VSS.
2: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION_REG<7>).

Four of PORTB's pins, RB7:RB4, have an interrupt on change feature. Only pins configured as inputs can cause this interrupt to occur (i.e. any RB7:RB4 pin configured as an output is excluded from the interrupt on change comparison). The input pins (of RB7:RB4) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB7:RB4 are OR'ed together to generate the RB Port Change Interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from SLEEP. The user, in the interrupt service routine, can clear the interrupt in the following manner:

a) Any read or write of PORTB. This will end the mismatch condition.
b) Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.

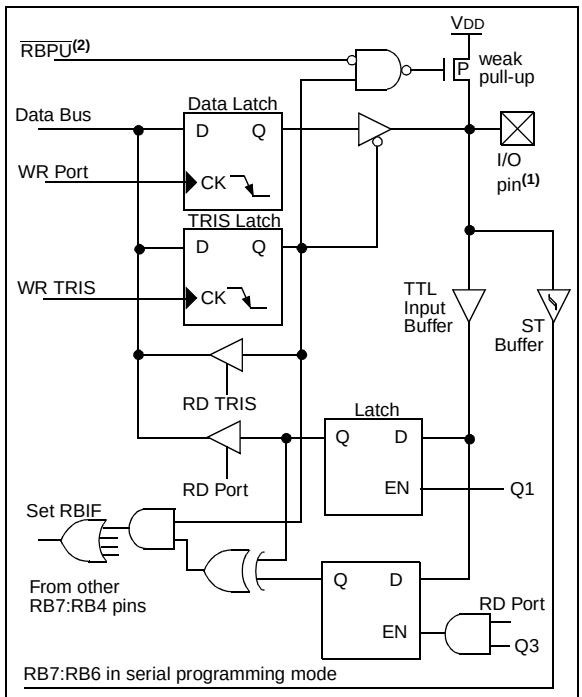
The interrupt on change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt on change feature. Polling of PORTB is not recommended while using the interrupt on change feature.

This interrupt on mismatch feature, together with software configureable pull-ups on these four pins, allow easy interface to a keypad and make it possible for wake-up on key-depression. Refer to the Embedded Control Handbook, *"Implementing Wake-Up on Key Stroke"* (AN552).

RB0/INT is an external interrupt input pin and is configured using the INTEDG bit (OPTION_REG<6>).

RB0/INT is discussed in detail in Section 12.10.1.

FIGURE 3-4: BLOCK DIAGRAM OF RB7:RB4 PINS



Note 1: I/O pins have diode protection to VDD and VSS.
2: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION_REG<7>).

Note: When using Low Voltage ICSP Programming (LVP) and the pull-ups on PORTB are enabled, bit 3 in the TRISB register must be cleared to disable the pull-up on RB3 and ensure the proper operation of the device.

TABLE 3-3: PORTB FUNCTIONS

Name	Bit#	Buffer	Function
RB0/INT	bit0	TTL/ST(1)	Input/output pin or external interrupt input. Internal software programmable weak pull-up.
RB1	bit1	TTL	Input/output pin. Internal software programmable weak pull-up.
RB2	bit2	TTL	Input/output pin. Internal software programmable weak pull-up.
RB3/PGM	bit3	TTL	Input/output pin or programming pin in LVP mode. Internal software programmable weak pull-up.
RB4	bit4	TTL	Input/output pin (with interrupt on change). Internal software programmable weak pull-up.
RB5	bit5	TTL	Input/output pin (with interrupt on change). Internal software programmable weak pull-up.
RB6/PGC	bit6	TTL/ST(2)	Input/output pin (with interrupt on change) or In-Circuit Debugger pin. Internal software programmable weak pull-up. Serial programming clock.
RB7/PGD	bit7	TTL/ST(2)	Input/output pin (with interrupt on change) or In-Circuit Debugger pin. Internal software programmable weak pull-up. Serial programming data.

Legend: TTL = TTL input, ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.

2: This buffer is a Schmitt Trigger input when used in serial programming mode.

TABLE 3-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
06h, 106h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
86h, 186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
81h, 181h	OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged. Shaded cells are not used by PORTB.

© 1999 Microchip Technology Inc.

3.3 PORTC and the TRISC Register

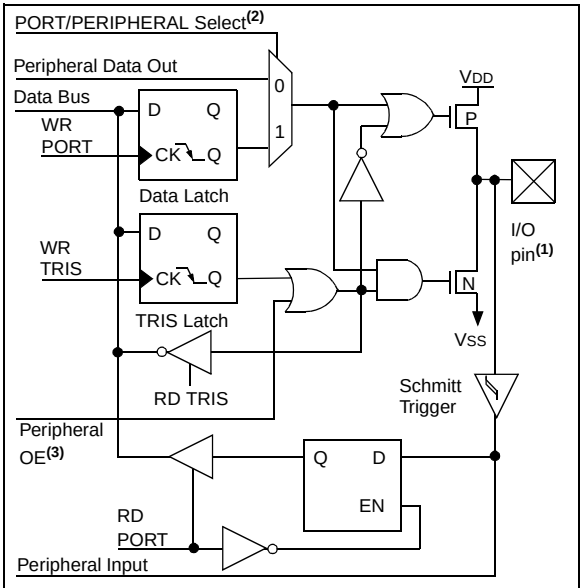
PORTC is an 8-bit wide, bi-directional port. The corresponding data direction register is TRISC. Setting a TRISC bit (=1) will make the corresponding PORTC pin an input (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISC bit (=0) will make the corresponding PORTC pin an output (i.e., put the contents of the output latch on the selected pin).

PORTC is multiplexed with several peripheral functions (Table 3-5). PORTC pins have Schmitt Trigger input buffers.

When the I^2C module is enabled, the PORTC (3:4) pins can be configured with normal I^2C levels or with SMBUS levels by using the CKE bit (SSPSTAT <6>).

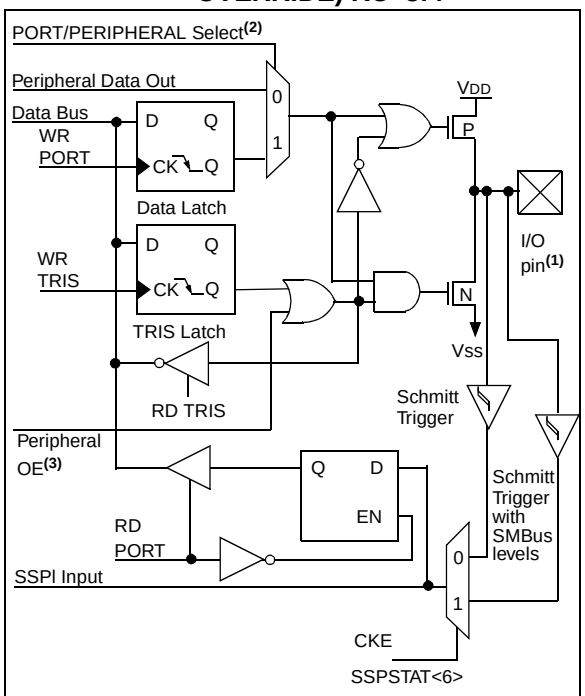
When enabling peripheral functions, care should be taken in defining TRIS bits for each PORTC pin. Some peripherals override the TRIS bit to make a pin an output, while other peripherals override the TRIS bit to make a pin an input. Since the TRIS bit override is in effect while the peripheral is enabled, read-modify-write instructions (`BSF, BCF, XORWF`) with TRISC as destination should be avoided. The user should refer to the corresponding peripheral section for the correct TRIS bit settings.

FIGURE 3-5: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE) RC<0:2> RC<5:7>



Note 1: I/O pins have diode protection to VDD and VSS.
2: Port/Peripheral select signal selects between port data and peripheral output.
3: Peripheral OE (output enable) is only activated if peripheral select is active.

FIGURE 3-6: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE) RC<3:4>



Note 1: I/O pins have diode protection to VDD and VSS.
2: Port/Peripheral select signal selects between port data and peripheral output.
3: Peripheral OE (output enable) is only activated if peripheral select is active.

TABLE 3-5: PORTC FUNCTIONS

Name	Bit#	Buffer Type	Function
RC0/T1OSO/T1CKI	bit0	ST	Input/output port pin or Timer1 oscillator output/Timer1 clock input
RC1/T1OSI/CCP2	bit1	ST	Input/output port pin or Timer1 oscillator input or Capture2 input/ Compare2 output/PWM2 output
RC2/CCP1	bit2	ST	Input/output port pin or Capture1 input/Compare1 output/PWM1 output
RC3/SCK/SCL	bit3	ST	RC3 can also be the synchronous serial clock for both SPI and I^2C modes.
RC4/SDI/SDA	bit4	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I^2C mode).
RC5/SDO	bit5	ST	Input/output port pin or Synchronous Serial Port data output
RC6/TX/CK	bit6	ST	Input/output port pin or USART Asynchronous Transmit or Synchronous Clock
RC7/RX/DT	bit7	ST	Input/output port pin or USART Asynchronous Receive or Synchronous Data

Legend: ST = Schmitt Trigger input

TABLE 3-6: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged.

© 1999 Microchip Technology Inc.

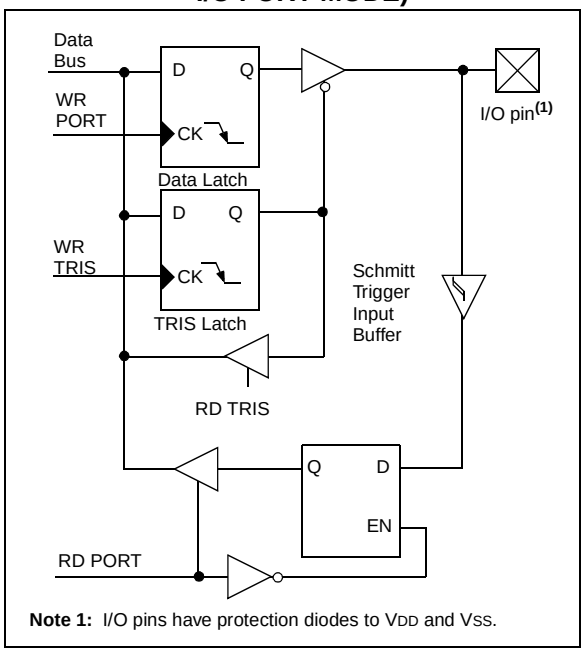
3.4 PORTD and TRISD Registers

This section is not applicable to the PIC16F873 or PIC16F876.

PORTD is an 8-bit port with Schmitt Trigger input buffers. Each pin is individually configurable as an input or output.

PORTD can be configured as an 8-bit wide microprocessor port (parallel slave port) by setting control bit PSPMODE (TRISE<4>). In this mode, the input buffers are TTL.

FIGURE 3-7: PORTD BLOCK DIAGRAM (IN I/O PORT MODE)



Note 1: I/O pins have protection diodes to VDD and VSS.

TABLE 3-7: PORTD FUNCTIONS

Name	Bit#	Buffer Type	Function
RD0/PSP0	bit0	ST/TTL(1)	Input/output port pin or parallel slave port bit0
RD1/PSP1	bit1	ST/TTL(1)	Input/output port pin or parallel slave port bit1
RD2/PSP2	bit2	ST/TTL(1)	Input/output port pin or parallel slave port bit2
RD3/PSP3	bit3	ST/TTL(1)	Input/output port pin or parallel slave port bit3
RD4/PSP4	bit4	ST/TTL(1)	Input/output port pin or parallel slave port bit4
RD5/PSP5	bit5	ST/TTL(1)	Input/output port pin or parallel slave port bit5
RD6/PSP6	bit6	ST/TTL(1)	Input/output port pin or parallel slave port bit6
RD7/PSP7	bit7	ST/TTL(1)	Input/output port pin or parallel slave port bit7

Legend: ST = Schmitt Trigger input TTL = TTL input

Note 1: Input buffers are Schmitt Triggers when in I/O mode and TTL buffer when in Parallel Slave Port Mode.

TABLE 3-8: SUMMARY OF REGISTERS ASSOCIATED WITH PORTD

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
08h	PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	uuuu uuuu
88h	TRISD	PORTD Data Direction Register								1111 1111	1111 1111
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by PORTD.

3.5 PORTE and TRISE Register

This section is not applicable to the PIC16F873 or PIC16F876.

PORTE has three pins, RE0/$\overline{RD}$/AN5, RE1/$\overline{WR}$/AN6 and RE2/$\overline{CS}$/AN7, which are individually configurable as inputs or outputs. These pins have Schmitt Trigger input buffers.

I/O PORTE becomes control inputs for the microprocessor port when bit PSPMODE (TRISE<4>) is set. In this mode, the user must make sure that the TRISE<2:0> bits are set (pins are configured as digital inputs). Ensure ADCON1 is configured for digital I/O. In this mode, the input buffers are TTL.

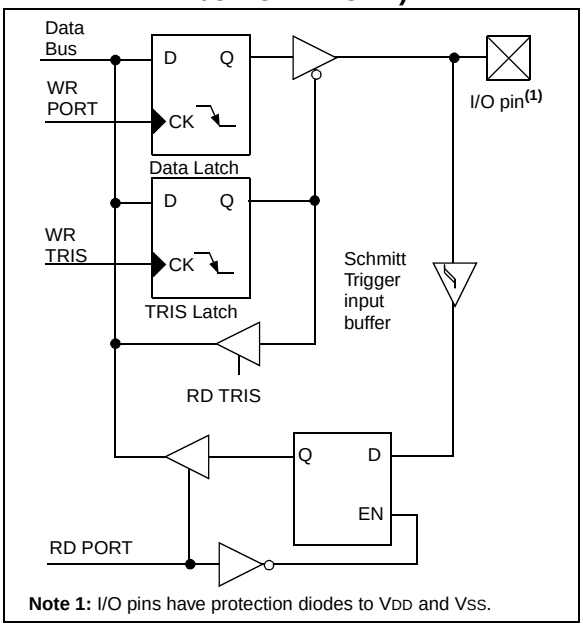
Register 3-1 shows the TRISE register, which also controls the parallel slave port operation.

PORTE pins are multiplexed with analog inputs. When selected as an analog input, these pins will read as '0's.

TRISE controls the direction of the RE pins, even when they are being used as analog inputs. The user must make sure to keep the pins configured as inputs when using them as analog inputs.

Note:	On a Power-on Reset, these pins are configured as analog inputs.

FIGURE 3-8: PORTE BLOCK DIAGRAM (IN I/O PORT MODE)



Note 1: I/O pins have protection diodes to VDD and VSS.

REGISTER 3-1: TRISE REGISTER (ADDRESS 89h)

R-0	R-0	R/W-0	R/W-0	U-0	R/W-1	R/W-1	R/W-1
IBF	OBF	IBOV	PSPMODE	—	bit2	bit1	bit0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n= Value at POR reset

Parallel Slave Port Status/Control Bits

bit 7 : **IBF:** Input Buffer Full Status bit
1 = A word has been received and is waiting to be read by the CPU
0 = No word has been received

bit 6: **OBF**: Output Buffer Full Status bit
1 = The output buffer still holds a previously written word
0 = The output buffer has been read

bit 5: **IBOV**: Input Buffer Overflow Detect bit (in microprocessor mode)
1 = A write occurred when a previously input word has not been read (must be cleared in software)
0 = No overflow occurred

bit 4: **PSPMODE**: Parallel Slave Port Mode Select bit
1 = Parallel slave port mode
0 = General purpose I/O mode

bit 3: **Unimplemented**: Read as '0'

PORTE Data Direction Bits

bit 2: **Bit2**: Direction Control bit for pin RE2/$\overline{CS}$/AN7
1 = Input
0 = Output

bit 1: **Bit1**: Direction Control bit for pin RE1/$\overline{WR}$/AN6
1 = Input
0 = Output

bit 0: **Bit0**: Direction Control bit for pin RE0/$\overline{RD}$/AN5
1 = Input
0 = Output

© 1999 Microchip Technology Inc.

TABLE 3-9: PORTE FUNCTIONS

Name	Bit#	Buffer Type	Function
RE0/$\overline{RD}$/AN5	bit0	ST/TTL(1)	Input/output port pin or read control input in parallel slave port mode or analog input: $\overline{RD}$ 1 = Not a read operation 0 = Read operation. Reads PORTD register (if chip selected)
RE1/$\overline{WR}$/AN6	bit1	ST/TTL(1)	Input/output port pin or write control input in parallel slave port mode or analog input: $\overline{WR}$ 1 = Not a write operation 0 = Write operation. Writes PORTD register (if chip selected)
RE2/$\overline{CS}$/AN7	bit2	ST/TTL(1)	Input/output port pin or chip select control input in parallel slave port mode or analog input: $\overline{CS}$ 1 = Device is not selected 0 = Device is selected

Legend: ST = Schmitt Trigger input TTL = TTL input

Note 1: Input buffers are Schmitt Triggers when in I/O mode and TTL buffers when in Parallel Slave Port Mode.

TABLE 3-10: SUMMARY OF REGISTERS ASSOCIATED WITH PORTE

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
09h	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by PORTE.

3.6 Parallel Slave Port

The Parallel Slave Port is not implemented on the PIC16F873 or PIC16F876.

PORTD operates as an 8-bit wide Parallel Slave Port or microprocessor port when control bit PSPMODE (TRISE<4>) is set. In slave mode, it is asynchronously readable and writable by the external world through $\overline{RD}$ control input pin RE0/$\overline{RD}$ and $\overline{WR}$ control input pin RE1/$\overline{WR}$.

It can directly interface to an 8-bit microprocessor data bus. The external microprocessor can read or write the PORTD latch as an 8-bit latch. Setting bit PSPMODE enables port pin RE0/$\overline{RD}$ to be the $\overline{RD}$ input, RE1/$\overline{WR}$ to be the $\overline{WR}$ input and RE2/$\overline{CS}$ to be the $\overline{CS}$ (chip select) input. For this functionality, the corresponding data direction bits of the TRISE register (TRISE<2:0>) must be configured as inputs (set). The A/D port configuration bits PCFG3:PCFG0 (ADCON1<3:0>) must be set to configure pins RE2:RE0 as digital I/O.

There are actually two 8-bit latches. One for data-out and one for data input. The user writes 8-bit data to the PORTD data latch and reads data from the port pin latch (note that they have the same address). In this mode, the TRISD register is ignored, since the microprocessor is controlling the direction of data flow.

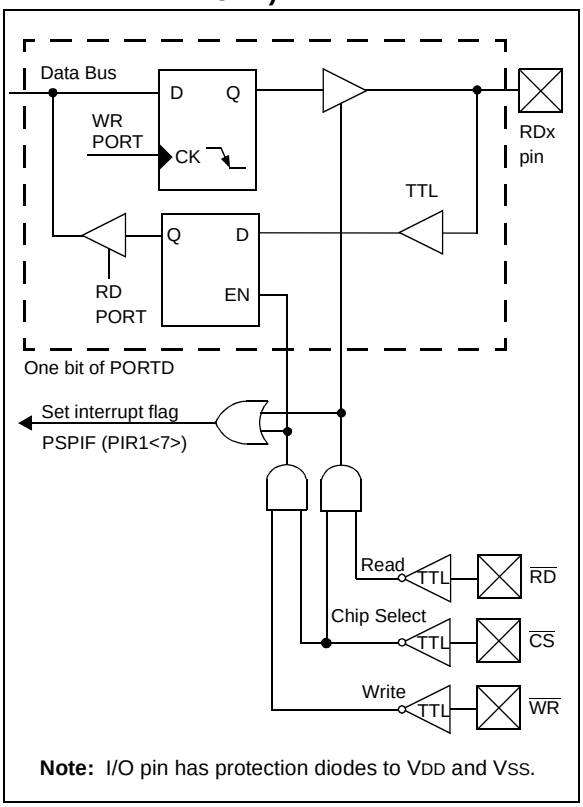
A write to the PSP occurs when both the $\overline{CS}$ and $\overline{WR}$ lines are first detected low. When either the $\overline{CS}$ or $\overline{WR}$ lines become high (level triggered), the Input Buffer Full (IBF) status flag bit (TRISE<7>) is set on the Q4 clock cycle, following the next Q2 cycle, to signal the write is complete (Figure 3-10). The interrupt flag bit PSPIF (PIR1<7>) is also set on the same Q4 clock cycle. IBF can only be cleared by reading the PORTD input latch. The Input Buffer Overflow (IBOV) status flag bit (TRISE<5>) is set if a second write to the PSP is attempted when the previous byte has not been read out of the buffer.

A read from the PSP occurs when both the $\overline{CS}$ and $\overline{RD}$ lines are first detected low. The Output Buffer Full (OBF) status flag bit (TRISE<6>) is cleared immediately (Figure 3-11) indicating that the PORTD latch is waiting to be read by the external bus. When either the $\overline{CS}$ or $\overline{RD}$ pin becomes high (level triggered), the interrupt flag bit PSPIF is set on the Q4 clock cycle, following the next Q2 cycle, indicating that the read is complete. OBF remains low until data is written to PORTD by the user firmware.

When not in PSP mode, the IBF and OBF bits are held clear. However, if flag bit IBOV was previously set, it must be cleared in firmware.

An interrupt is generated and latched into flag bit PSPIF when a read or write operation is completed. PSPIF must be cleared by the user in firmware and the interrupt can be disabled by clearing the interrupt enable bit PSPIE (PIE1<7>).

FIGURE 3-9: PORTD AND PORTE BLOCK DIAGRAM (PARALLEL SLAVE PORT)



Note: I/O pin has protection diodes to VDD and VSS.

© 1999 Microchip Technology Inc.

FIGURE 3-10: PARALLEL SLAVE PORT WRITE WAVEFORMS

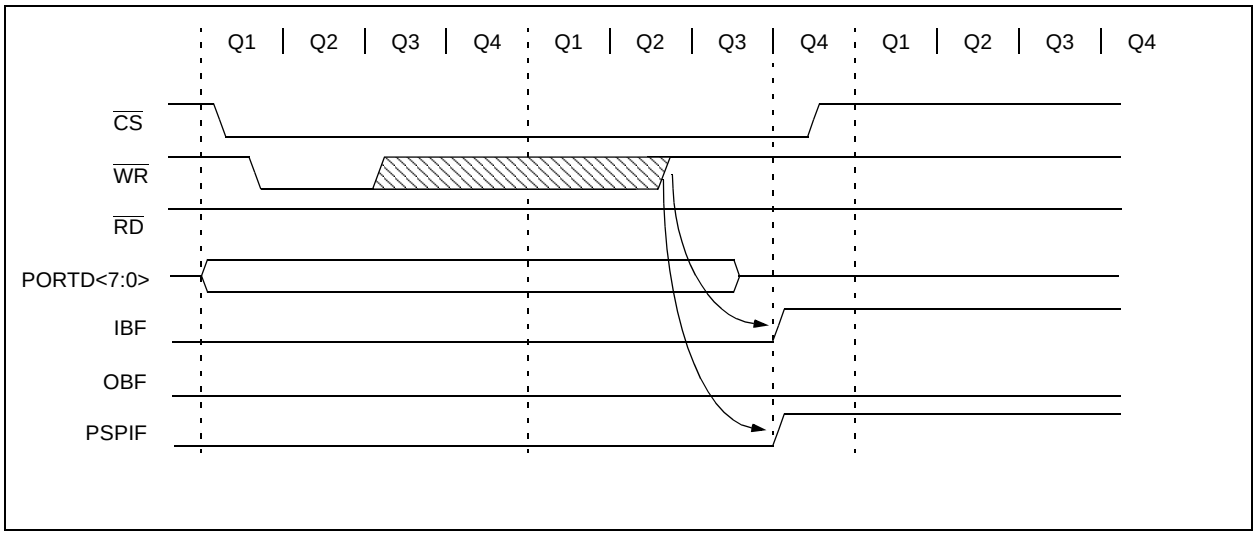


FIGURE 3-11: PARALLEL SLAVE PORT READ WAVEFORMS

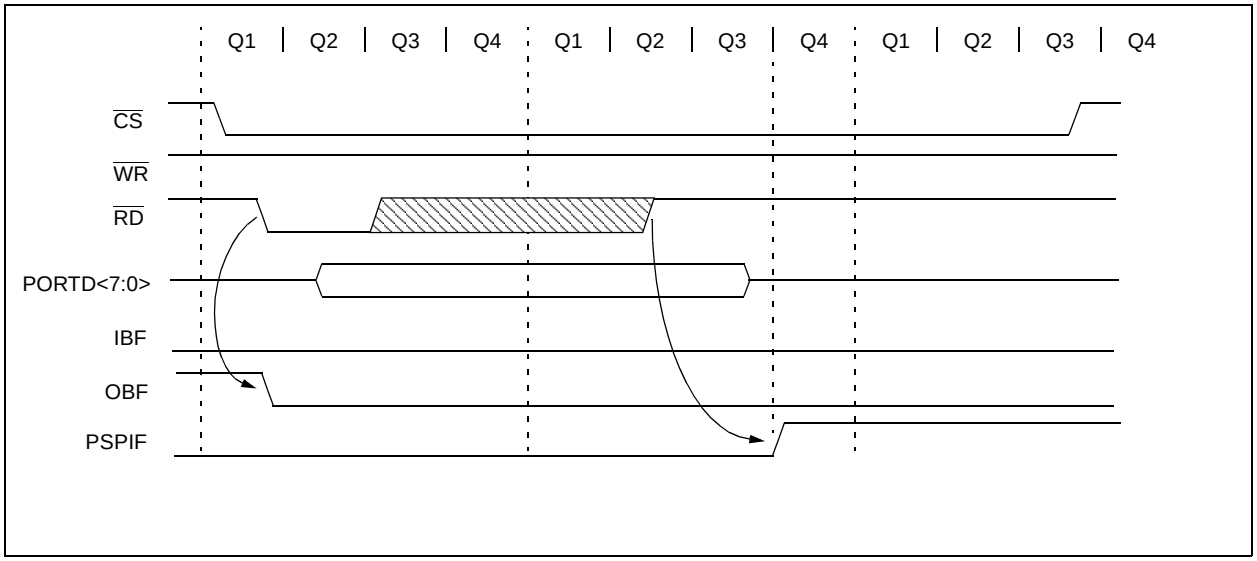


TABLE 3-11: REGISTERS ASSOCIATED WITH PARALLEL SLAVE PORT

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
08h	PORTD	Port data latch when written: Port pins when read								xxxx xxxx	uuuu uuuu
09h	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111
0Ch	PIR1	PSPIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by the Parallel Slave Port.

NOTES:

© 1999 Microchip Technology Inc.

4.0 DATA EEPROM AND FLASH PROGRAM MEMORY

The Data EEPROM and FLASH Program Memory are readable and writable during normal operation over the entire VDD range. A bulk erase operation may not be issued from user code (which includes removing code protection). The data memory is not directly mapped in the register file space. Instead it is indirectly addressed through the Special Function Registers (SFR).

There are six SFRs used to read and write the program and data EEPROM memory. These registers are:

- EECON1
- EECON2
- EEDATA
- EEDATH
- EEADR
- EEADRH

The EEPROM data memory allows byte read and write. When interfacing to the data memory block, EEDATA holds the 8-bit data for read/write and EEADR holds the address of the EEPROM location being accessed. The registers EEDATH and EEADRH are not used for data EEPROM access. These devices have up to 256 bytes of data EEPROM with an address range from 0h to FFh.

The EEPROM data memory is rated for high erase/write cycles. The write time is controlled by an on-chip timer. The write time will vary with voltage and temperature, as well as from chip-to-chip. Please refer to the specifications for exact limits.

The program memory allows word reads and writes. Program memory access allows for checksum calculation and calibration table storage. A byte or word write automatically erases the location and writes the new data (erase before write). Writing to program memory will cease operation until the write is complete. The program memory cannot be accessed during the write, therefore code cannot execute. During the write operation, the oscillator continues to clock the peripherals, and therefore they continue to operate. Interrupt events will be detected and essentially “queued” until the write is completed. When the write completes, the next instruction in the pipeline is executed and the branch to the interrupt vector address will occur.

When interfacing to the program memory block, the EEDATH:EEDATA registers form a two byte word, which holds the 14-bit data for read/write. The EEADRH:EEADR registers form a two byte word, which holds the 13-bit address of the EEPROM location being accessed. These devices can have up to 8K words of program EEPROM with an address range from 0h to 3FFFh. The unused upper bits in both the EEDATH and EEDATA registers all read as “0’s”.

The value written to program memory does not need to be a valid instruction. Therefore, up to 14-bit numbers can be stored in memory for use as calibration parameters, serial numbers, packed 7-bit ASCII, etc. Executing a program memory location containing data that forms an invalid instruction results in a NOP.

4.1 EEADR

The address registers can address up to a maximum of 256 bytes of data EEPROM or up to a maximum of 8K words of program FLASH.

When selecting a program address value, the MSByte of the address is written to the EEADRH register and the LSByte is written to the EEADR register. When selecting a data address value, only the LSByte of the address is written to the EEADR register.

On the PIC16F873/874 devices with 128 bytes of EEPROM, the MSbit of the EEADR must always be cleared to prevent inadvertent access to the wrong location. This also applies to the program memory. The upper MSbits of EEADRH must always be clear.

4.2 EECON1 and EECON2 Registers

EECON1 is the control register for memory accesses.

EECON2 is not a physical register. Reading EECON2 will read all '0's. The EECON2 register is used exclusively in the memory write sequence.

Control bit EEPGD determines if the access will be a program or a data memory access. When clear, any subsequent operations will operate on the data memory. When set, any subsequent operations will operate on the program memory.

Control bits RD and WR initiate read and write operations, respectively. These bits cannot be cleared, only set, in software. They are cleared in hardware at the completion of the read or write operation. The inability to clear the WR bit in software prevents the accidental or premature termination of a write operation.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear. The WRERR bit is set when a write operation is interrupted by a $\overline{\text{MCLR}}$ reset or a WDT time-out reset during normal operation. In these situations, following reset, the user can check the WRERR bit and rewrite the location. The value of the data and address registers and the EEPGD bit remains unchanged.

Interrupt flag bit EEIF, in the PIR2 register, is set when write is complete. It must be cleared in software.

© 1999 Microchip Technology Inc.

REGISTER 4-1: EECON1 REGISTER (ADDRESS 18Ch)

R/W-x	U-0	U-0	U-0	R/W-x	R/W-0	R/S-0	R/S-0
EEPGD	—	—	—	WRERR	WREN	WR	RD
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n= Value at POR reset

bit 7: **EEPGD**: Program / Data EEPROM Select bit
1 = Accesses Program memory
0 = Accesses data memory
(This bit cannot be changed while a read or write operation is in progress)

bit 6:4: **Unimplemented:** Read as '0'

bit 3: **WRERR**: EEPROM Error Flag bit
1 = A write operation is prematurely terminated
(any MCLR reset or any WDT reset during normal operation)
0 = The write operation completed

bit 2: **WREN**: EEPROM Write Enable bit
1 = Allows write cycles
0 = Inhibits write to the EEPROM

bit 1: **WR**: Write Control bit
1 = initiates a write cycle. (The bit is cleared by hardware once write is complete. The WR bit can only be set (not cleared) in software.
0 = Write cycle to the EEPROM is complete

bit 0: **RD**: Read Control bit
1 = Initiates an EEPROM read RD is cleared in hardware. The RD bit can only be set (not cleared) in software.
0 = Does not initiate an EEPROM read

© 1999 Microchip Technology Inc.

4.3 Reading the Data EEPROM Memory

To read a data memory location, the user must write the address to the EEADR register, clear the EEPGD control bit (EECON1<7>) and then set control bit RD (EECON1<0>). The data is available in the very next instruction cycle of the EEDATA register, therefore it can be read by the next instruction. EEDATA will hold this value until another read operation or until it is written to by the user (during a write operation).

EXAMPLE 4-1: DATA EEPROM READ

```
BSF   STATUS, RP1  ;
BCF   STATUS, RP0  ;Bank 2
MOVLW DATA_EE_ADDR ;
MOVWF EEADR        ;Data Memory Address to read
BSF   STATUS, RP0  ;Bank 3
BCF   EECON1, EEPGD;Point to DATA memory
BSF   EECON1, RD   ;EEPROM Read
BCF   STATUS, RP0  ;Bank 2
MOVF  EEDATA, W    ;W = EEDATA
```

4.4 Writing to the Data EEPROM Memory

To write an EEPROM data location, the address must first be written to the EEADR register and the data written to the EEDATA register. Then the sequence in Example 4-2 must be followed to initiate the write cycle.

EXAMPLE 4-2: DATA EEPROM WRITE

```
                BSF     STATUS, RP1   ;
                BCF     STATUS, RP0   ; Bank 2
                MOVLW   DATA_EE_ADDR  ;
                MOVWF   EEADR         ; Data Memory Address to write
                MOVLW   DATA_EE_DATA  ;
                MOVWF   EEDATA        ; Data Memory Value to write
                BSF     STATUS, RP0   ; Bank 3
                BCF     EECON1, EEPGD ; Point to DATA memory
                BSF     EECON1, WREN  ; Enable writes

                BCF     INTCON, GIE   ; Disable Interrupts
                MOVLW   55h           ;
Required        MOVWF   EECON2        ; Write 55h
Sequence        MOVLW   AAh           ;
                MOVWF   EECON2        ; Write AAh
                BSF     EECON1, WR    ; Set WR bit to begin write
                BSF     INTCON, GIE   ; Enable Interrupts

                SLEEP                 ; Wait for interrupt to signal write complete
                BCF     EECON1, WREN  ; Disable writes
```

The write will not initiate if the above sequence is not exactly followed (write 55h to EECON2, write AAh to EECON2, then set WR bit) for each byte. It is strongly recommended that interrupts be disabled during this code segment.

Additionally, the WREN bit in EECON1 must be set to enable writes. This mechanism prevents accidental writes to data EEPROM due to unexpected code execution (i.e., runaway programs). The WREN bit should be kept clear at all times, except when updating the EEPROM. The WREN bit is not cleared by hardware

After a write sequence has been initiated, clearing the WREN bit will not affect the current write cycle. The WR bit will be inhibited from being set unless the WREN bit is set. The WREN bit must be set on a previous instruction. Both WR and WREN cannot be set with the same instruction.

At the completion of the write cycle, the WR bit is cleared in hardware and the EEPROM Write Complete Interrupt Flag bit (EEIF) is set. EEIF must be cleared by software.

4.5 Reading the FLASH Program Memory

A program memory location may be read by writing two bytes of the address to the EEADR and EEADRH registers, setting the EEPGD control bit (EECON1<7>) and then setting control bit RD (EECON1<0>). Once the read control bit is set, the microcontroller will use the next two instruction cycles to read the data. The data is available in the EEDATA and EEDATH registers after the second NOP instruction. Therefore, it can be read as two bytes in the following instructions. The EEDATA and EEDATH registers will hold this value until another read operation or until it is written to by the user (during a write operation).

EXAMPLE 4-3: FLASH PROGRAM READ

```
                BSF     STATUS, RP1      ;
                BCF     STATUS, RP0      ; Bank 2
                MOVLW   ADDRH            ;
                MOVWF   EEADRH           ; MSByte of Program Address to read
                MOVLW   ADDRL            ;
                MOVWF   EEADR            ; LSByte of Program Address to read
                BSF     STATUS, RP0      ; Bank 3
                BSF     EECON1, EEPGD    ; Point to PROGRAM memory
Required        BSF     EECON1, RD       ; EEPROM Read
Sequence
                NOP                     ; memory is read in the next two cycles after BSF EECON1,RD
                NOP                      ;

                BCF     STATUS, RP0      ; Bank 2


                MOVF    EEDATA, W        ; W = LSByte of Program EEDATA
                MOVF    EEDATH, W        ; W = MSByte of Program EEDATA
```


© 1999 Microchip Technology Inc.

4.6 Writing to the FLASH Program Memory

A word of the FLASH program memory may only be written to if the word is in a non-code protected segment of memory and the WRT configuration bit is set. To write a FLASH program location, the first two bytes of the address must be written to the EEADR and EEADRH registers and two bytes of the data to the EEDATA and EEDATH registers, set the EEPGD control bit (EECON1<7>), and then set control bit WR (EECON1<1>). The sequence in Example 4-4 must be followed to initiate a write to program memory.

The microcontroller will then halt internal operations during the next two instruction cycles for the TPEW (parameter D133) in which the write takes place. This is not SLEEP mode, as the clocks and peripherals will continue to run. Therefore, the two instructions following the "BSF EECON, WR" should be NOP instructions. After the write cycle, the microcontroller will resume operation with the 3rd instruction after the EECON1 write instruction.

EXAMPLE 4-4: FLASH PROGRAM WRITE

```
                BSF     STATUS, RP1      ;
                BCF     STATUS, RP0      ; Bank 2
                MOVLW   ADDRH            ;
                MOVWF   EEADRH           ; MSByte of Program Address to read
                MOVLW   ADDRL            ;
                MOVWF   EEADR            ; LSByte of Program Address to read
                MOVLW   DATAH            ;
                MOVWF   EEDATH           ; MS Program Memory Value to write
                MOVLW   DATAL            ;
                MOVWF   EEDATA           ; LS Program Memory Value to write
                BSF     STATUS, RP0      ; Bank 3
                BSF     EECON1, EEPGD    ; Point to PROGRAM memory
                BSF     EECON1, WREN     ; Enable writes

                BCF     INTCON, GIE      ; Disable Interrupts
                MOVLW   55h              ;
Required        MOVWF   EECON2           ; Write 55h
Sequence        MOVLW   AAh              ;
                MOVWF   EECON2           ; Write AAh
                BSF     EECON1, WR       ; Set WR bit to begin write


                NOP                     ; Instructions here are ignored by the microcontroller
                NOP
                                        ; Microcontroller will halt operation and wait for
                                        ; a write complete. After the write
                                        ; the microcontroller continues with 3rd instruction
                BSF    INTCON,  GIE     ; Enable Interrupts
                BCF    EECON1, WREN     ; Disable writes
```

4.7 Write Verify

Depending on the application, good programming practice may dictate that the value written to the memory should be verified against the original value. This should be used in applications where excessive writes can stress bits near the specification limit.

Generally a write failure will be a bit which was written as a '1', but reads back as a '0' (due to leakage off the bit).

4.8 Protection Against Spurious Write

4.8.1 EEPROM DATA MEMORY

There are conditions when the device may not want to write to the data EEPROM memory. To protect against spurious EEPROM writes, various mechanisms have been built-in. On power-up, the WREN bit is cleared. Also, the Power-up Timer (72 ms duration) prevents EEPROM write.

The write initiate sequence and the WREN bit together help prevent an accidental write during brown-out, power glitch, or software malfunction.

4.8.2 PROGRAM FLASH MEMORY

To protect against spurious writes to FLASH program memory, the WRT bit in the configuration word may be programmed to '0' to prevent writes. The write initiate sequence must also be followed. WRT and the configuration word cannot be programmed by user code, only through the use of an external programmer.

4.9 Operation during Code Protect

Each reprogrammable memory block has its own code protect mechanism. External Read and Write operations are disabled if either of these mechanisms are enabled.

4.9.1 DATA EEPROM MEMORY

The microcontroller itself can both read and write to the internal Data EEPROM, regardless of the state of the code protect configuration bit.

4.9.2 PROGRAM FLASH MEMORY

The microcontroller can read and execute instructions out of the internal FLASH program memory, regardless of the state of the code protect configuration bits. However the WRT configuration bit and the code protect bits have different effects on writing to program memory. Table 4-1 shows the various configurations and status of reads and writes. To erase the WRT or code protection bits in the configuration word requires that the device be fully erased.

TABLE 4-1: READ/WRITE STATE OF INTERNAL FLASH PROGRAM MEMORY

Configuration Bits			Memory Location	Internal Read	Internal Write	ICSP Read	ICSP Write
CP1	CP0	WRT					
0	0	x	All program memory	Yes	No	No	No
0	1	0	Unprotected areas	Yes	No	Yes	No
0	1	0	Protected areas	Yes	No	No	No
0	1	1	Unprotected areas	Yes	Yes	Yes	No
0	1	1	Protected areas	Yes	No	No	No
1	0	0	Unprotected areas	Yes	No	Yes	No
1	0	0	Protected areas	Yes	No	No	No
1	0	1	Unprotected areas	Yes	Yes	Yes	No
1	0	1	Protected areas	Yes	No	No	No
1	1	0	All program memory	Yes	No	Yes	Yes
1	1	1	All program memory	Yes	Yes	Yes	Yes

TABLE 4-2: REGISTERS ASSOCIATED WITH DATA EEPROM/PROGRAM FLASH

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
10Dh	EEADR	EEPROM address register								xxxx xxxx	uuuu uuuu
10Fh	EEADRH	—	—	—	EEPROM address high					xxxx xxxx	uuuu uuuu
10Ch	EEDATA	EEPROM data resister								xxxx xxxx	uuuu uuuu
10Eh	EEDATH	—	—	EEPROM data resister high						xxxx xxxx	uuuu uuuu
18Ch	EECON1	EEPGD	—	—	—	WRERR	WREN	WR	RD	x--- x000	x--- u000
18Dh	EECON2	EEPROM control resister2 (not a physical resister)									
8Dh	PIE2	—	(1)	—	EEIE	BCLIE	—	—	CCP2IE	-r-0 0--0	-r-0 0--0
0Dh	PIR2	—	(1)	—	EEIF	BCLIF	—	—	CCP2IF	-r-0 0--0	-r-0 0--0

Legend: x = unknown, u = unchanged, r = reserved, - = unimplemented read as '0'. Shaded cells are not used during FLASH/EEPROM access.

Note 1: These bits are reserved; always maintain these bits clear.

© 1999 Microchip Technology Inc.

5.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select
- Interrupt on overflow from FFh to 00h
- Edge select for external clock

Figure 5-1 is a block diagram of the Timer0 module and the prescaler shared with the WDT.

Additional information on the Timer0 module is available in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).

Timer mode is selected by clearing bit T0CS (OPTION_REG<5>). In timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If the TMR0 register is written, the increment is inhibited for the following two instruction cycles. The user can work around this by writing an adjusted value to the TMR0 register.

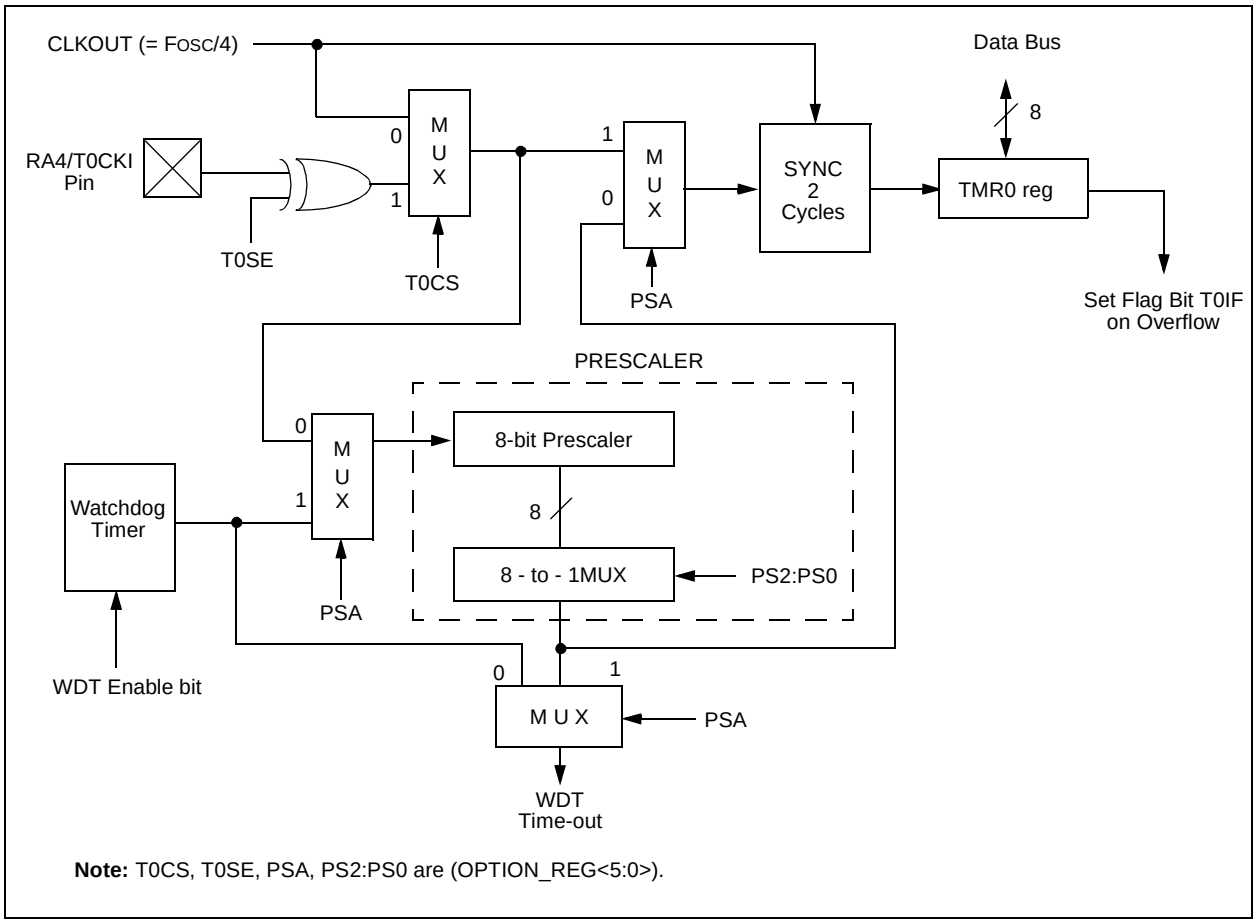
Counter mode is selected by setting bit T0CS (OPTION_REG<5>). In counter mode, Timer0 will increment either on every rising or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the Timer0 Source Edge Select bit T0SE (OPTION_REG<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed in detail in Section 5.2.

The prescaler is mutually exclusively shared between the Timer0 module and the watchdog timer. The prescaler is not readable or writable. Section 5.3 details the operation of the prescaler.

5.1 Timer0 Interrupt

The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h. This overflow sets bit T0IF (INTCON<2>). The interrupt can be masked by clearing bit T0IE (INTCON<5>). Bit T0IF must be cleared in software by the Timer0 module interrupt service routine before re-enabling this interrupt. The TMR0 interrupt cannot awaken the processor from SLEEP since the timer is shut off during SLEEP.

FIGURE 5-1: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



Note: T0CS, T0SE, PSA, PS2:PS0 are (OPTION_REG<5:0>).

5.2 Using Timer0 with an External Clock

When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks. Therefore, it is necessary for T0CKI to be high for at least 2Tosc (and a small RC delay of 20 ns) and low for at least 2Tosc (and a small RC delay of 20 ns). Refer to the electrical specification of the desired device.

5.3 Prescaler

There is only one prescaler available, which is mutually exclusively shared between the Timer0 module and the watchdog timer. A prescaler assignment for the Timer0 module means that there is no prescaler for the watchdog timer, and vice-versa. This prescaler is not readable or writable (see Figure 5-1).

The PSA and PS2:PS0 bits (OPTION_REG<3:0>) determine the prescaler assignment and prescale ratio.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g. `CLRF 1, MOVWF 1, BSF 1,` x....etc.) will clear the prescaler. When assigned to WDT, a `CLRWDT` instruction will clear the prescaler along with the Watchdog Timer. The prescaler is not readable or writable.

Note:	Writing to TMR0, when the prescaler is assigned to Timer0, will clear the prescaler count, but will not change the prescaler assignment.

REGISTER 5-1: OPTION_REG REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **$\overline{\text{RBPU}}$**

bit 6: **INTEDG**

bit 5: **T0CS**: TMR0 Clock Source Select bit
`1` = Transition on T0CKI pin
`0` = Internal instruction cycle clock (CLKOUT)

bit 4: **T0SE**: TMR0 Source Edge Select bit
`1` = Increment on high-to-low transition on T0CKI pin
`0` = Increment on low-to-high transition on T0CKI pin

bit 3: **PSA**: Prescaler Assignment bit
`1` = Prescaler is assigned to the WDT
`0` = Prescaler is assigned to the Timer0 module

bit 2-0: **PS2:PS0**: Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

Note:	To avoid an unintended device RESET, the instruction sequence shown in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023) must be executed when changing the prescaler assignment from Timer0 to the WDT. This sequence must be followed even if the WDT is disabled.

© 1999 Microchip Technology Inc.

TABLE 5-1: REGISTERS ASSOCIATED WITH TIMER0

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
01h,101h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
81h,181h	OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by Timer0.

NOTES:

© 1999 Microchip Technology Inc.

6.0 TIMER1 MODULE

The Timer1 module is a 16-bit timer/counter consisting of two 8-bit registers (TMR1H and TMR1L), which are readable and writable. The TMR1 Register pair (TMR1H:TMR1L) increments from 0000h to FFFFh and rolls over to 0000h. The TMR1 Interrupt, if enabled, is generated on overflow, which is latched in interrupt flag bit TMR1IF (PIR1<0>). This interrupt can be enabled/disabled by setting/clearing TMR1 interrupt enable bit TMR1IE (PIE1<0>).

Timer1 can operate in one of two modes:

- As a timer
- As a counter

The operating mode is determined by the clock select bit, TMR1CS (T1CON<1>).

In timer mode, Timer1 increments every instruction cycle. In counter mode, it increments on every rising edge of the external clock input.

Timer1 can be enabled/disabled by setting/clearing control bit TMR1ON (T1CON<0>).

Timer1 also has an internal “reset input”. This reset can be generated by either of the two CCP modules (Section 8.0). Register 6-1 shows the Timer1 control register.

When the Timer1 oscillator is enabled (T1OSCEN is set), the RC1/T1OSI/CCP2 and RC0/T1OSO/T1CKI pins become inputs. That is, the TRISC<1:0> value is ignored.

Additional information on timer modules is available in the PICmicro™ Mid-range MCU Family Reference Manual (DS33023).

REGISTER 6-1: T1CON: TIMER1 CONTROL REGISTER (ADDRESS 10h)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7-6: **Unimplemented:** Read as ’0’

bit 5-4: **T1CKPS1:T1CKPS0**: Timer1 Input Clock Prescale Select bits
11 = 1:8 Prescale value
10 = 1:4 Prescale value
01 = 1:2 Prescale value
00 = 1:1 Prescale value

bit 3: **T1OSCEN**: Timer1 Oscillator Enable Control bit
1 = Oscillator is enabled
0 = Oscillator is shut off (The oscillator inverter is turned off to eliminate power drain)

bit 2: **$\overline{\text{T1SYNC}}$**: Timer1 External Clock Input Synchronization Control bit

TMR1CS = 1
1 = Do not synchronize external clock input
0 = Synchronize external clock input

TMR1CS = 0
This bit is ignored. Timer1 uses the internal clock when TMR1CS = 0.

bit 1: **TMR1CS**: Timer1 Clock Source Select bit
1 = External clock from pin RC0/T1OSO/T1CKI (on the rising edge)
0 = Internal clock (FOSC/4)

bit 0: **TMR1ON**: Timer1 On bit
1 = Enables Timer1
0 = Stops Timer1

6.1 Timer1 Operation in Timer Mode

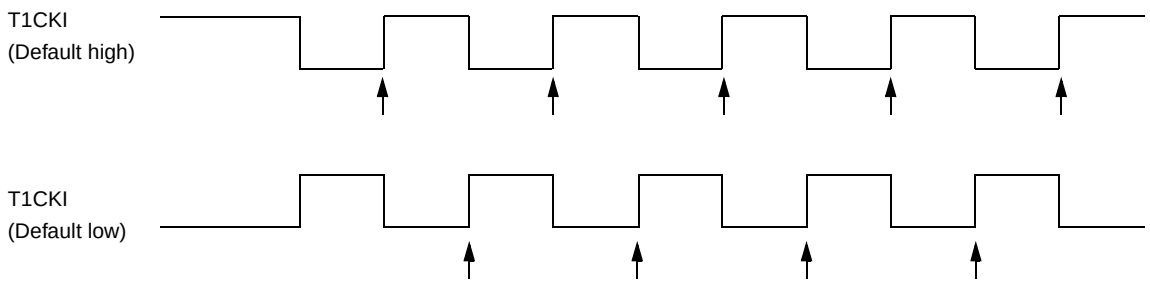
Timer mode is selected by clearing the TMR1CS (T1CON<1>) bit. In this mode, the input clock to the timer is FOSC/4. The synchronize control bit $\overline{\text{T1SYNC}}$ (T1CON<2>) has no effect since the internal clock is always in sync.

6.2 Timer1 Counter Operation

Timer1 may operate in asynchronous or usynchronous mode depnding on the setting of the TMR1CS bit.

When Timer1 is being incremented via an external source, increments occur on a rising edge. After Timer1 is enabled in counter mode, the module must first have a falling edge before the counter begins to increment.

FIGURE 6-1: TIMER1 INCREMENTING EDGE



Note: Arrows indicate counter increments.

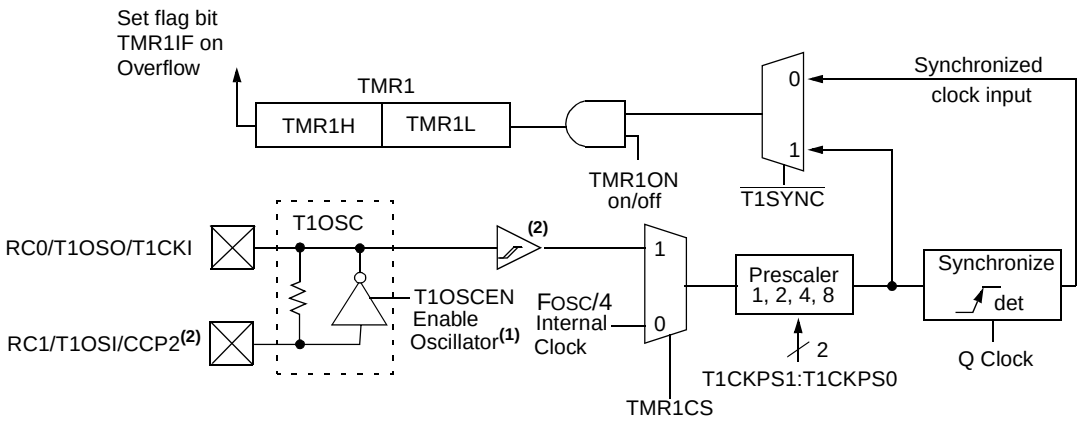
6.3 Timer1 Operation in Synchronized Counter Mode

Counter mode is selected by setting bit TMR1CS. In this mode, the timer increments on every rising edge of clock input on pin RC1/T1OSI/CCP2, when bit T1OSCEN is set, or on pin RC0/T1OSO/T1CKI, when bit T1OSCEN is cleared.

If $\overline{\text{T1SYNC}}$ is cleared, then the external clock input is synchronized with internal phase clocks. The synchronization is done after the prescaler stage. The prescaler stage is an asynchronous ripple-counter.

In this configuration, during SLEEP mode, Timer1 will not increment even if the external clock is present, since the synchronization circuit is shut off. The prescaler however will continue to increment.

FIGURE 6-2: TIMER1 BLOCK DIAGRAM



Note 1: When the T1OSCEN bit is cleared, the inverter is turned off. This eliminates power drain.
2: For the PIC16F873/876, the Schmitt Trigger is not implemented in external clock mode.

© 1999 Microchip Technology Inc.

6.4 Timer1 Operation in Asynchronous Counter Mode

If control bit $\overline{\text{T1SYNC}}$ (T1CON<2>) is set, the external clock input is not synchronized. The timer continues to increment asynchronous to the internal phase clocks. The timer will continue to run during SLEEP and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (Section 6.4.1).

In asynchronous counter mode, Timer1 can not be used as a time-base for capture or compare operations.

6.4.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L while the timer is running from an external asynchronous clock will guarantee a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the timer register.

Reading the 16-bit value requires some care. Examples 12-2 and 12-3 in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023) show how to read and write Timer1 when it is running in asynchronous mode.

6.5 Timer1 Oscillator

A crystal oscillator circuit is built-in between pins T1OSI (input) and T1OSO (amplifier output). It is enabled by setting control bit T1OSCEN (T1CON<3>). The oscillator is a low power oscillator rated up to 200 kHz. It will continue to run during SLEEP. It is primarily intended for use with a 32 kHz crystal. Table 6-1 shows the capacitor selection for the Timer1 oscillator.

The Timer1 oscillator is identical to the LP oscillator. The user must provide a software time delay to ensure proper oscillator start-up.

TABLE 6-1: CAPACITOR SELECTION FOR THE TIMER1 OSCILLATOR

Osc Type	Freq	C1	C2
LP	32 kHz	33 pF	33 pF
	100 kHz	15 pF	15 pF
	200 kHz	15 pF	15 pF
These values are for design guidance only.			
Crystals Tested:			
32.768 kHz	Epson C-001R32.768K-A		± 20 PPM
100 kHz	Epson C-2 100.00 KC-P		± 20 PPM
200 kHz	STD XTL 200.000 kHz		± 20 PPM

Note 1: Higher capacitance increases the stability of oscillator but also increases the start-up time.

2: Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.

6.6 Resetting Timer1 using a CCP Trigger Output

If the CCP1 or CCP2 module is configured in compare mode to generate a "special event trigger" (CCP1M3:CCP1M0 = 1011), this signal will reset Timer1.

Note: The special event triggers from the CCP1 and CCP2 modules will not set interrupt flag bit TMR1IF (PIR1<0>).

Timer1 must be configured for either timer or synchronized counter mode to take advantage of this feature. If Timer1 is running in asynchronous counter mode, this reset operation may not work.

In the event that a write to Timer1 coincides with a special event trigger from CCP1 or CCP2, the write will take precedence.

In this mode of operation, the CCPRxH:CCPRxL register pair effectively becomes the period register for Timer1.

6.7 Resetting of Timer1 Register Pair (TMR1H, TMR1L)

TMR1H and TMR1L registers are not reset to 00h on a POR or any other reset except by the CCP1 and CCP2 special event triggers.

T1CON register is reset to 00h on a Power-on Reset or a Brown-out Reset, which shuts off the timer and leaves a 1:1 prescale. In all other resets, the register is unaffected.

6.8 Timer1 Prescaler

The prescaler counter is cleared on writes to the TMR1H or TMR1L registers.

TABLE 6-2: REGISTERS ASSOCIATED WITH TIMER1 AS A TIMER/COUNTER

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
0Bh,8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	--00 0000	--uu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by the Timer1 module.

Note 1: Bits PSPIE and PSPIF are reserved on the PIC16F873/874; always maintain these bits clear.

© 1999 Microchip Technology Inc.

7.0 TIMER2 MODULE

Timer2 is an 8-bit timer with a prescaler and a postscaler. It can be used as the PWM time-base for the PWM mode of the CCP module(s). The TMR2 register is readable and writable, and is cleared on any device reset.

The input clock (FOSC/4) has a prescale option of 1:1, 1:4 or 1:16, selected by control bits T2CKPS1:T2CKPS0 (T2CON<1:0>).

The Timer2 module has an 8-bit period register PR2. Timer2 increments from 00h until it matches PR2 and then resets to 00h on the next increment cycle. PR2 is a readable and writable register. The PR2 register is initialized to FFh upon reset.

The match output of TMR2 goes through a 4-bit postscaler (which gives a 1:1 to 1:16 scaling inclusive) to generate a TMR2 interrupt (latched in flag bit TMR2IF, (PIR1<1>)).

Timer2 can be shut off by clearing control bit TMR2ON (T2CON<2>) to minimize power consumption.

Register 7-1 shows the Timer2 control register.

Additional information on timer modules is available in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).

7.1 Timer2 Prescaler and Postscaler

The prescaler and postscaler counters are cleared when any of the following occurs:

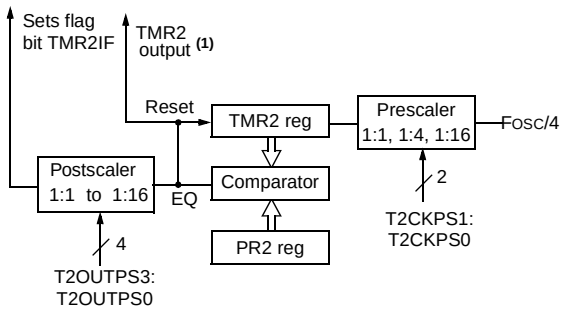
- a write to the TMR2 register
- a write to the T2CON register
- any device reset (POR, $\overline{MCLR}$ reset, WDT reset or BOR)

TMR2 is not cleared when T2CON is written.

7.2 Output of TMR2

The output of TMR2 (before the postscaler) is fed to the SSPort module, which optionally uses it to generate shift clock.

FIGURE 7-1: TIMER2 BLOCK DIAGRAM



Note 1: TMR2 register output can be software selected by the SSP module as a baud clock.

REGISTER 7-1: T2CON: TIMER2 CONTROL REGISTER (ADDRESS 12h)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **Unimplemented:** Read as '0'

bit 6-3: **TOUTPS3:TOUTPS0**: Timer2 Output Postscale Select bits
0000 = 1:1 Postscale
0001 = 1:2 Postscale
0010 = 1:3 Postscale
•
•
•
1111 = 1:16 Postscale

bit 2: **TMR2ON**: Timer2 On bit
1 = Timer2 is on
0 = Timer2 is off

bit 1-0: **T2CKPS1:T2CKPS0**: Timer2 Clock Prescale Select bits
00 = Prescaler is 1
01 = Prescaler is 4
1x = Prescaler is 16

TABLE 7-1: REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
92h	PR2	Timer2 Period Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by the Timer2 module.
Note 1: Bits PSPIE and PSPIF are reserved on the PIC16F873/874; always maintain these bits clear.

© 1999 Microchip Technology Inc.

8.0 CAPTURE/COMPARE/PWM MODULES

Each Capture/Compare/PWM (CCP) module contains a 16-bit register which can operate as a:

- 16-bit Capture register
- 16-bit Compare register
- PWM master/slave Duty Cycle register

Both the CCP1 and CCP2 modules are identical in operation, with the exception being the operation of the special event trigger. Table 8-1 and Table 8-2 show the resources and interactions of the CCP module(s). In the following sections, the operation of a CCP module is described with respect to CCP1. CCP2 operates the same as CCP1, except where noted.

CCP1 Module:

Capture/Compare/PWM Register1 (CCPR1) is comprised of two 8-bit registers: CCPR1L (low byte) and CCPR1H (high byte). The CCP1CON register controls the operation of CCP1. The special event trigger is generated by a compare match and will reset Timer1.

CCP2 Module:

Capture/Compare/PWM Register1 (CCPR2) is comprised of two 8-bit registers: CCPR2L (low byte) and CCPR2H (high byte). The CCP2CON register controls the operation of CCP2. The special event trigger is generated by a compare match and will reset Timer1 and start an A/D conversion (if the A/D module is enabled).

Additional information on CCP modules is available in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023) and in Application Note 594, “Using the CCP Modules” (DS00594).

TABLE 8-1: CCP MODE - TIMER RESOURCES REQUIRED

CCP Mode	Timer Resource
Capture	Timer1
Compare	Timer1
PWM	Timer2

TABLE 8-2: INTERACTION OF TWO CCP MODULES

CCPx Mode	CCPy Mode	Interaction
Capture	Capture	Same TMR1 time-base.
Capture	Compare	The compare should be configured for the special event trigger, which clears TMR1.
Compare	Compare	The compare(s) should be configured for the special event trigger, which clears TMR1.
PWM	PWM	The PWMs will have the same frequency and update rate (TMR2 interrupt).
PWM	Capture	None.
PWM	Compare	None.

REGISTER 8-1: CCP1CON REGISTER/CCP2CON REGISTER (ADDRESS: 17h/1dh)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CCPxX	CCPxY	CCPxM3	CCPxM2	CCPxM1	CCPxM0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-6: **Unimplemented:** Read as '0'

bit 5-4: **CCPxX:CCPxY**: PWM Least Significant bits
Capture Mode: Unused
Compare Mode: Unused
PWM Mode: These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPRxL.

bit 3-0: **CCPxM3:CCPxM0**: CCPx Mode Select bits
0000 = Capture/Compare/PWM off (resets CCPx module)
0100 = Capture mode, every falling edge
0101 = Capture mode, every rising edge
0110 = Capture mode, every 4th rising edge
0111 = Capture mode, every 16th rising edge
1000 = Compare mode, set output on match (CCPxIF bit is set)
1001 = Compare mode, clear output on match (CCPxIF bit is set)
1010 = Compare mode, generate software interrupt on match (CCPxIF bit is set, CCPx pin is unaffected)
1011 = Compare mode, trigger special event (CCPxIF bit is set, CCPx pin is unaffected); CCP1 resets TMR1; CCP2 resets TMR1 and starts an A/D conversion (if A/D module is enabled)
11xx = PWM mode

© 1999 Microchip Technology Inc.

8.1 Capture Mode

In Capture mode, CCPR1H:CCPR1L captures the 16-bit value of the TMR1 register when an event occurs on pin RC2/CCP1. An event is defined as:

- Every falling edge
- Every rising edge
- Every 4th rising edge
- Every 16th rising edge

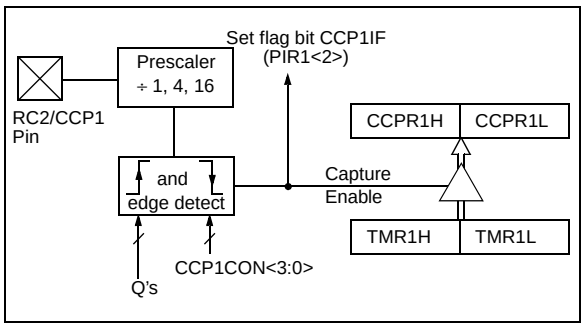
An event is selected by control bits CCP1M3:CCP1M0 (CCP1CON<3:0>). When a capture is made, the interrupt request flag bit CCP1IF (PIR1<2>) is set. The interrupt flag must be cleared in software. If another capture occurs before the value in register CCPR1 is read, the old captured value will be lost.

8.1.1 CCP PIN CONFIGURATION

In Capture mode, the RC2/CCP1 pin should be configured as an input by setting the TRISC<2> bit.

Note:	If the RC2/CCP1 pin is configured as an output, a write to the port can cause a capture condition.

FIGURE 8-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



8.1.2 TIMER1 MODE SELECTION

Timer1 must be running in timer mode or synchronized counter mode for the CCP module to use the capture feature. In asynchronous counter mode, the capture operation may not work.

8.1.3 SOFTWARE INTERRUPT

When the capture mode is changed, a false capture interrupt may be generated. The user should keep bit CCP1IE (PIE1<2>) clear to avoid false interrupts and should clear the flag bit CCP1IF following any such change in operating mode.

8.1.4 CCP PRESCALER

There are four prescaler settings, specified by bits CCP1M3:CCP1M0. Whenever the CCP module is turned off, or the CCP module is not in capture mode, the prescaler counter is cleared. Any reset will clear the prescaler counter.

Switching from one capture prescaler to another may generate an interrupt. Also, the prescaler counter will not be cleared, therefore, the first capture may be from a non-zero prescaler. Example 8-1 shows the recommended method for switching between capture prescalers. This example also clears the prescaler counter and will not generate the “false” interrupt.

EXAMPLE 8-1: CHANGING BETWEEN CAPTURE PRESCALERS

```
CLRF    CCP1CON      ;Turn CCP module off
MOVLW   NEW_CAPT_PS  ;Load the W reg with
                     ; the new precscaler
                     ; move value and CCP ON
MOVWF   CCP1CON      ;Load CCP1CON with this
                     ; value
```

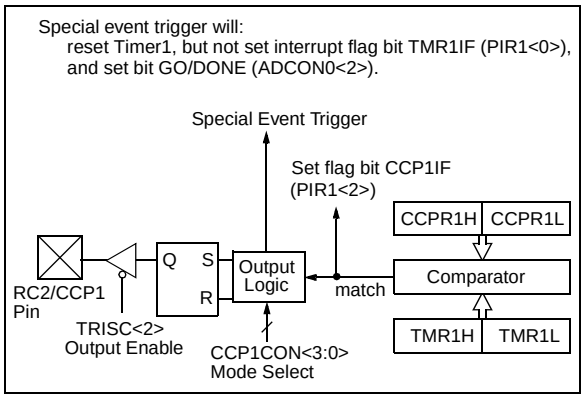
8.2 Compare Mode

In Compare mode, the 16-bit CCPR1 register value is constantly compared against the TMR1 register pair value. When a match occurs, the RC2/CCP1 pin is:

- Driven high
- Driven low
- Remains unchanged

The action on the pin is based on the value of control bits CCP1M3:CCP1M0 (CCP1CON<3:0>). At the same time, interrupt flag bit CCP1IF is set.

FIGURE 8-2: COMPARE MODE OPERATION BLOCK DIAGRAM



8.2.1 CCP PIN CONFIGURATION

The user must configure the RC2/CCP1 pin as an output by clearing the TRISC<2> bit.

Note:	Clearing the CCP1CON register will force the RC2/CCP1 compare output latch to the default low level. This is not the data latch.

8.2.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode if the CCP module is using the compare feature. In Asynchronous Counter mode, the compare operation may not work.

8.2.3 SOFTWARE INTERRUPT MODE

When Generate Software Interrupt mode is chosen, the CCP1 pin is not affected. The CCPIF bit is set causing a CCP interrupt (if enabled).

8.2.4 SPECIAL EVENT TRIGGER

In this mode, an internal hardware trigger is generated, which may be used to initiate an action.

The special event trigger output of CCP1 resets the TMR1 register pair. This allows the CCPR1 register to effectively be a 16-bit programmable period register for Timer1.

The special event trigger output of CCP2 resets the TMR1 register pair and starts an A/D conversion (if the A/D module is enabled).

Note:	The special event trigger from the CCP1and CCP2 modules will not set interrupt flag bit TMR1IF (PIR1<0>).

8.3 PWM Mode (PWM)

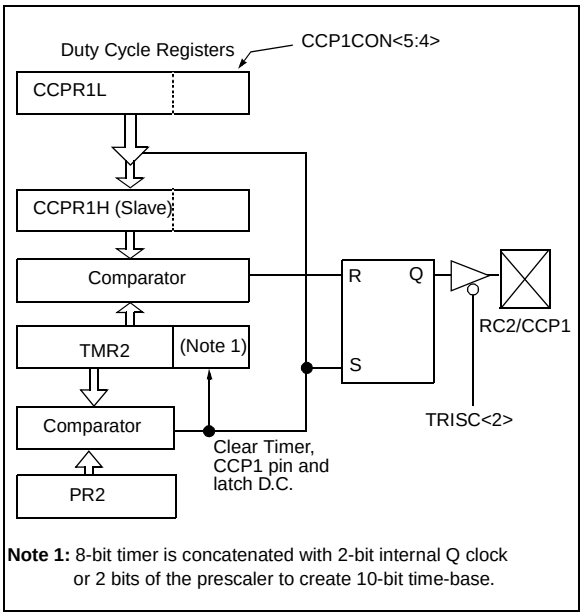
In pulse width modulation mode, the CCPx pin produces up to a 10-bit resolution PWM output. Since the CCP1 pin is multiplexed with the PORTC data latch, the TRISC<2> bit must be cleared to make the CCP1 pin an output.

Note:	Clearing the CCP1CON register will force the CCP1 PWM output latch to the default low level. This is not the PORTC I/O data latch.

Figure 8-3 shows a simplified block diagram of the CCP module in PWM mode.

For a step-by-step procedure on how to set up the CCP module for PWM operation, see Section 8.3.3.

FIGURE 8-3: SIMPLIFIED PWM BLOCK DIAGRAM

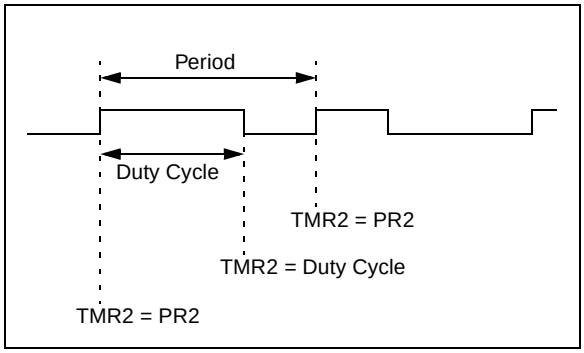


Note 1: 8-bit timer is concatenated with 2-bit internal Q clock or 2 bits of the prescaler to create 10-bit time-base.

© 1999 Microchip Technology Inc.

A PWM output (Figure 8-4) has a time-base (period) and a time that the output stays high (duty cycle). The frequency of the PWM is the inverse of the period (1/period).

FIGURE 8-4: PWM OUTPUT



8.3.1 PWM PERIOD

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following formula:

$$\text{PWM period} = [(PR2) + 1] \cdot 4 \cdot T_{OSC} \cdot (\text{TMR2 prescale value})$$

PWM frequency is defined as 1 / [PWM period].

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The CCP1 pin is set (exception: if PWM duty cycle = 0%, the CCP1 pin will not be set)
- The PWM duty cycle is latched from CCPR1L into CCPR1H

Note:	The Timer2 postscaler (see Section 8.1) is not used in the determination of the PWM frequency. The postscaler could be used to have a servo update rate at a different frequency than the PWM output.

8.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the CCPR1L register and to the CCP1CON<5:4> bits. Up to 10-bit resolution is available. The CCPR1L contains the eight MSbs and the CCP1CON<5:4> contains the two LSbs. This 10-bit value is represented by CCPR1L:CCP1CON<5:4>. The following equation is used to calculate the PWM duty cycle in time:

$$\text{PWM duty cycle} = (\text{CCPR1L:CCP1CON<5:4>}) \cdot Tosc \cdot (\text{TMR2 prescale value})$$

CCPR1L and CCP1CON<5:4> can be written to at any time, but the duty cycle value is not latched into CCPR1H until after a match between PR2 and TMR2 occurs (i.e., the period is complete). In PWM mode, CCPR1H is a read-only register.

The CCPR1H register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

When the CCPR1H and 2-bit latch match TMR2 concatenated with an internal 2-bit Q clock or 2 bits of the TMR2 prescaler, the CCP1 pin is cleared.

Maximum PWM resolution (bits) for a given PWM frequency:

$$\text{Resolution} = \frac{\log\left(\frac{F_{OSC}}{F_{PWM}}\right)}{\log(2)} \text{ bits}$$

Note:	If the PWM duty cycle value is longer than the PWM period, the CCP1 pin will not be cleared.

8.3.3 SET-UP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for PWM operation:

1. Set the PWM period by writing to the PR2 register.
2. Set the PWM duty cycle by writing to the CCPR1L register and CCP1CON<5:4> bits.
3. Make the CCP1 pin an output by clearing the TRISC<2> bit.
4. Set the TMR2 prescale value and enable Timer2 by writing to T2CON.
5. Configure the CCP1 module for PWM operation.

TABLE 8-3: REGISTERS ASSOCIATED WITH CAPTURE, COMPARE, AND TIMER1

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh	PIR2	—	—	—	—	—	—	—	CCP2IF	---- ---0	---- ---0
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh	PIE2	—	—	—	—	—	—	—	CCP2IE	---- ---0	---- ---0
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	--00 0000	--uu uuuu
15h	CCPR1L	Capture/Compare/PWM register1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM register1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
1Bh	CCPR2L	Capture/Compare/PWM register2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM register2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by Capture and Timer1.
Note 1: The PSP is not implemented on the PIC16F873/876; always maintain these bits clear.

TABLE 8-4: REGISTERS ASSOCIATED WITH PWM AND TIMER2

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh	PIR2	—	—	—	—	—	—	—	CCP2IF	---- ---0	---- ---0
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh	PIE2	—	—	—	—	—	—	—	CCP2IE	---- ---0	---- ---0
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
92h	PR2	Timer2 module's period register								1111 1111	1111 1111
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
15h	CCPR1L	Capture/Compare/PWM register1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM register1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
1Bh	CCPR2L	Capture/Compare/PWM register2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM register2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by PWM and Timer2.
Note 1: Bits PSPIE and PSPIF are reserved on the PIC16F873/876; always maintain these bits clear.

© 1999 Microchip Technology Inc.

9.0 MASTER SYNCHRONOUS SERIAL PORT (MSSP) MODULE

The Master Synchronous Serial Port (MSSP) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D converters, etc. The MSSP module can operate in one of two modes:

- Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I^2C)

Figure 9-1 shows a block diagram for the SPI mode, while Figure 9-5 and Figure 9-9 show the block diagrams for the two different I^2C modes of operation.

REGISTER 9-1: SSPSTAT: SYNC SERIAL PORT STATUS REGISTER (ADDRESS: 94h)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **SMP:** Sample bit
SPI Master Mode
1 = Input data sampled at end of data output time
0 = Input data sampled at middle of data output time
SPI Slave Mode
SMP must be cleared when SPI is used in slave mode
In I^2C master or slave mode:
1= Slew rate control disabled for standard speed mode (100 kHz and 1 MHz)
0= Slew rate control enabled for high speed mode (400 kHz)

bit 6: **CKE**: SPI Clock Edge Select (Figure 9-4, Figure 9-5 and Figure 9-6)
SPI Mode:
CKP = 0
1 = Transmit happens on transistion from active clock state to idle clock state
0 = Transmit happens on transistion from idle clock state to active clock state
CKP = 1
1 = Data transmitted on falling edge of SCK
0 = Data transmitted on rising edge of SCK
In I^2C Master or Slave Mode:
1 = Input levels conform to SMBUS spec
0 = Input levels conform to I^2C specs

bit 5: **D/A**: Data/$\overline{\text{Address}}$ bit (I^2C mode only)
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4: **P**: Stop bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared)
1 = Indicates that a stop bit has been detected last (this bit is '0' on RESET)
0 = Stop bit was not detected last

bit 3: **S**: Start bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared)
1 = Indicates that a start bit has been detected last (this bit is '0' on RESET)
0 = Start bit was not detected last

bit 2: **R/W**: Read/Write bit information (I^2C mode only)
This bit holds the R/W bit information following the last address match. This bit is only valid from the address match to the next start bit, stop bit or not $\overline{\text{ACK}}$ bit.
In I^2C slave mode:
1 = Read
0 = Write
In I^2C master mode:
1 = Transmit is in progress
0 = Transmit is not in progress.
Or'ing this bit with SEN, RSEN, PEN, RCEN or ACKEN will indicate if the MSSP is in IDLE mode.

bit 1: **UA**: Update Address (10-bit I^2C mode only)
1 = Indicates that the user needs to update the address in the SSPADD register
0 = Address does not need to be updated

bit 0: **BF**: Buffer Full Status bit
Receive (SPI and I^2C modes)
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty
Transmit (I^2C mode only)
1 = Data Transmit in progress (does not include the $\overline{\text{ACK}}$ and stop bits), SSPBUF is full
0 = Data Transmit complete (does not include the $\overline{\text{ACK}}$ and stop bits), SSPBUF is empty

© 1999 Microchip Technology Inc.

REGISTER 9-2: SSPCON: SYNC SERIAL PORT CONTROL REGISTER (ADDRESS 14h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **WCOL**: Write Collision Detect bit
Master Mode:
1 = A write to SSPBUF was attempted while the I^2C conditions were not valid
0 = No collision
Slave Mode:
1 = SSPBUF register is written while still transmitting the previous word (must be cleared in software)
0 = No collision

bit 6: **SSPOV**: Receive Overflow Indicator bit
In SPI mode
1 = A new byte is received while SSPBUF holds previous data. Data in SSPSR is lost on overflow. . In slave mode the user must read the SSPBUF, even if only transmitting data, to avoid overflows. In master mode the overflow bit is not set since each operation is initiated by writing to the SSPBUF register. (Must be cleared in software).
0 = No overflow
In I^2C mode
1 = A byte is received while the SSPBUF is holding the previous byte. SSPOV is a "don't care" in transmit mode. (Must be cleared in software).
0 = No overflow

bit 5: **SSPEN**: Synchronous Serial Port Enable bit
In SPI mode, when enabled, these pins must be properly configured as input or output.
1 = Enables serial port and configures SCK, SDO, SDI, and $\overline{SS}$ as the source of the serial port pins
0 = Disables serial port and configures these pins as I/O port pins
In I^2C mode, when enabled, these pins must be properly configured as input or output.
1 = Enables the serial port and configures the SDA and SCL pins as the source of the serial port pins
0 = Disables serial port and configures these pins as I/O port pins

bit 4: **CKP**: Clock Polarity Select bit
In SPI mode
1 = Idle state for clock is a high level
0 = Idle state for clock is a low level
In I^2C slave mode, SCK release control
1 = Enable clock
0 = Holds clock low (clock stretch) (Used to ensure data setup time)
In I^2C master mode
Unused in this mode

bit 3-0: **SSPM3:SSPM0**: Synchronous Serial Port Mode Select bits
0000 = SPI master mode, clock = FOSC/4
0001 = SPI master mode, clock = FOSC/16
0010 = SPI master mode, clock = FOSC/64
0011 = SPI master mode, clock = TMR2 output/2
0100 = SPI slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.
0101 = SPI slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin
0110 = I^2C slave mode, 7-bit address
0111 = I^2C slave mode, 10-bit address
1000 = I^2C master mode, clock = FOSC / (4 * (SSPADD+1))
1011 = I^2C firmware controlled master mode (slave idle)
1110 = I^2C firmware controlled master mode, 7-bit address with start and stop bit interrupts enabled
1111 = I^2C firmware controlled master mode, 10-bit address with start and stop bit interrupts enabled.
1001, 1010, 1100, 1101 = reserved

REGISTER 9-3: SSPCON2: SYNC SERIAL PORT CONTROL REGISTER2 (ADDRESS 91h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, Read as '0'
- n = Value at POR reset

bit 7: **GCEN**: General Call Enable bit (In I^2C slave mode only)
1 = Enable interrupt when a general call address (0000h) is received in the SSPSR.
0 = General call address disabled.

bit 6: **ACKSTAT**: Acknowledge Status bit (In I^2C master mode only)
In master transmit mode:
1 = Acknowledge was not received from slave
0 = Acknowledge was received from slave

bit 5: **ACKDT**: Acknowledge Data bit (In I^2C master mode only)
In master receive mode:
Value that will be transmitted when the user initiates an Acknowledge sequence at the end of a receive.
1 = Not Acknowledge
0 = Acknowledge

bit 4: **ACKEN**: Acknowledge Sequence Enable bit (In I^2C master mode only).
In master receive mode:
1 = Initiate Acknowledge sequence on SDA and SCL pins, and transmit ACKDT data bit. Automatically cleared by hardware.
0 = Acknowledge sequence idle

bit 3: **RCEN**: Receive Enable bit (In I^2C master mode only).
1 = Enables Receive mode for I^2C
0 = Receive idle

bit 2: **PEN**: Stop Condition Enable bit (In I^2C master mode only).
SCK release control
1 = Initiate Stop condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Stop condition idle

bit 1: **RSEN**: Repeated Start Condition Enabled bit (In I^2C master mode only)
1 = Initiate Repeated Start condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Repeated Start condition idle.

bit 0: **SEN**: Start Condition Enabled bit (In I^2C master mode only)
1 = Initiate Start condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Start condition idle.

Note: For bits ACKEN, RCEN, PEN, RSEN, SEN: If the I^2C module is not in the idle mode, this bit may not be set (no spooling), and the SSPBUF may not be written (or writes to the SSPBUF are disabled).

© 1999 Microchip Technology Inc.

9.1 SPI Mode

The SPI mode allows 8 bits of data to be synchronously transmitted and received simultaneously. All four modes of SPI are supported. To accomplish communication, typically three pins are used:

- Serial Data Out (SDO)
- Serial Data In (SDI)
- Serial Clock (SCK)

Additionally, a fourth pin may be used when in a slave mode of operation:

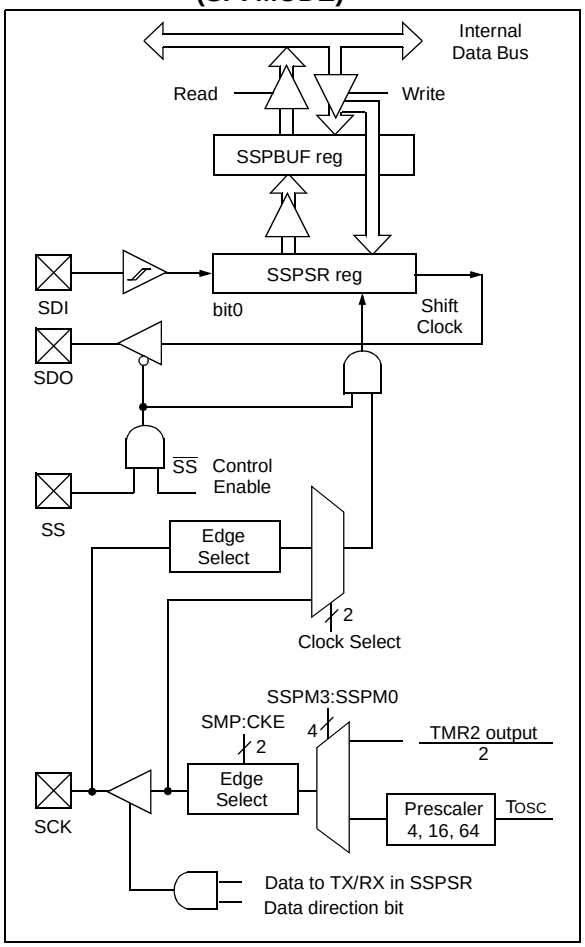
- Slave Select ($\overline{SS}$)

When initializing the SPI, several options need to be specified. This is done by programming the appropriate control bits (SSPCON<5:0> and SSPSTAT<7:6>). These control bits allow the following to be specified:

- Master Mode (SCK is the clock output)
- Slave Mode (SCK is the clock input)
- Clock Polarity (Idle state of SCK)
- Data input sample phase (middle or end of data output time)
- Clock edge (output data on rising/falling edge of SCK)
- Clock Rate (Master mode only)
- Slave Select Mode (Slave mode only)

Figure 9-4 shows the block diagram of the MSSP module when in SPI mode.

FIGURE 9-1: MSSP BLOCK DIAGRAM (SPI MODE)



To enable the serial port, MSSP Enable bit, SSPEN (SSPCON<5>) must be set. To reset or reconfigure SPI mode, clear bit SSPEN, re-initialize the SSPCON registers, and then set bit SSPEN. This configures the SDI, SDO, SCK and $\overline{SS}$ pins as serial port pins. For the pins to behave as the serial port function, some must have their data direction bits (in the TRIS register) appropriately programmed. That is:

- SDI is automatically controlled by the SPI module
- SDO must have TRISC<5> cleared
- SCK (Master mode) must have TRISC<3> cleared
- SCK (Slave mode) must have TRISC<3> set
- $\overline{SS}$ must have TRISA<5> set

Any serial port function that is not desired may be overridden by programming the corresponding data direction (TRIS) register to the opposite value.

9.1.1 MASTER MODE

The master can initiate the data transfer at any time because it controls the SCK. The master determines when the slave (Processor 2, Figure 9-5) is to broadcast data by the software protocol.

In master mode, the data is transmitted/received as soon as the SSPBUF register is written to. If the SPI module is only going to receive, the SDO output could be disabled (programmed as an input). The SSPSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPBUF register as if a normal received byte (interrupts and status bits appropriately set). This could be useful in receiver applications as a “line activity monitor”.

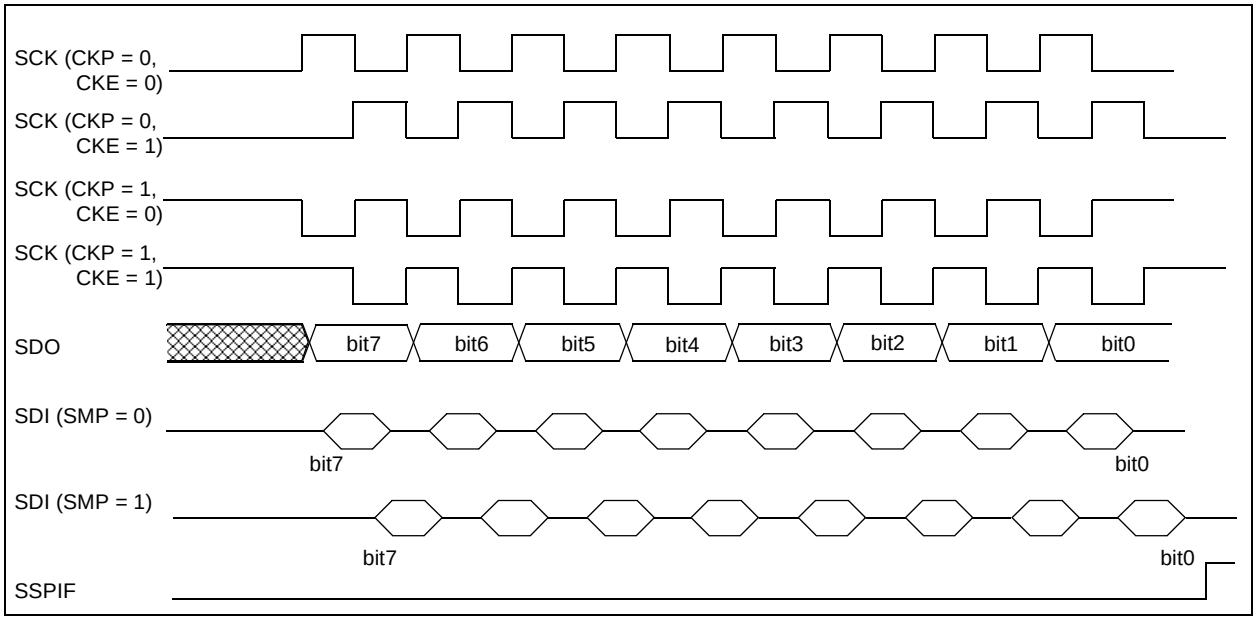
The clock polarity is selected by appropriately programming bit CKP (SSPCON<4>). This then would give waveforms for SPI communication as shown in Figure 9-6, Figure 9-8 and Figure 9-9 where the MSb is transmitted first. In master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- FOSC/4 (or TCY)
- FOSC/16 (or 4 • TCY)
- FOSC/64 (or 16 • TCY)
- Timer2 output/2

This allows a maximum bit clock frequency (at 20 MHz) of 5.0 MHz.

Figure 9-6 shows the waveforms for Master mode. When CKE = 1, the SDO data is valid before there is a clock edge on SCK. The change of the input sample is shown based on the state of the SMP bit. The time when the SSPBUF is loaded with the received data is shown.

FIGURE 9-2: SPI MODE TIMING, MASTER MODE



© 1999 Microchip Technology Inc.

9.1.2 SLAVE MODE

In slave mode, the data is transmitted and received as the external clock pulses appear on SCK. When the last bit is latched, the interrupt flag bit SSPIF (PIR1<3>) is set.

While in slave mode, the external clock is supplied by the external clock source on the SCK pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

While in sleep mode, the slave can transmit/receive data. When a byte is received, the device will wake-up from sleep.

Note:	When the SPI module is in Slave Mode with $\overline{SS}$ pin control enabled, (SSPCON<3:0> = 0100) the SPI module will reset if the $\overline{SS}$ pin is set to VDD.
Note:	If the SPI is used in Slave Mode with CKE = '1', then $\overline{SS}$ pin control must be enabled.

FIGURE 9-3: SPI MODE TIMING (SLAVE MODE WITH CKE = 0)

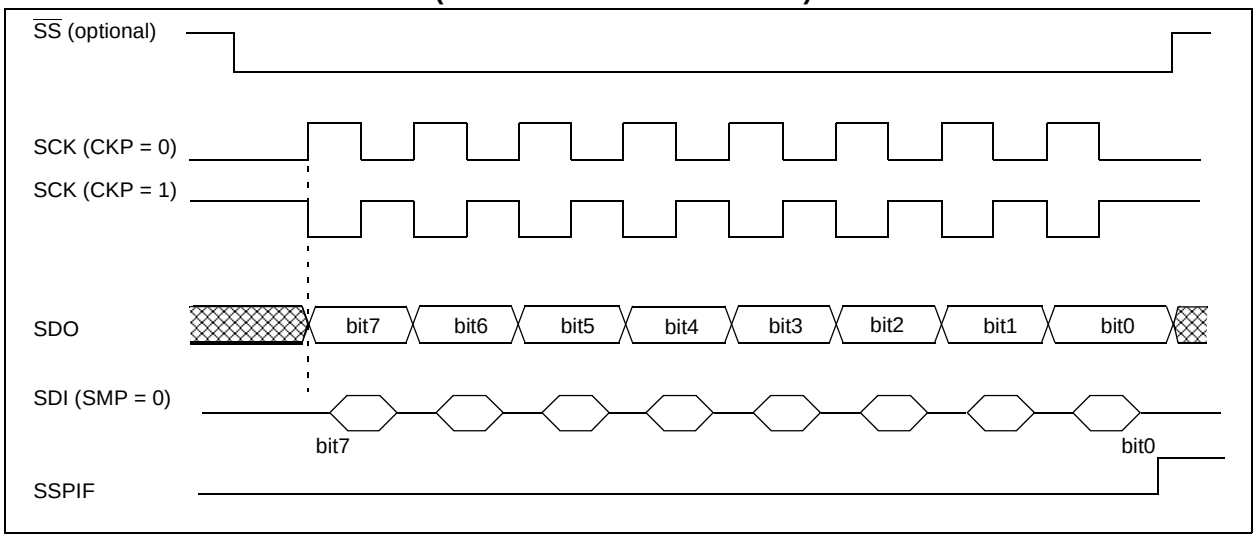


FIGURE 9-4: SPI MODE TIMING (SLAVE MODE WITH CKE = 1)

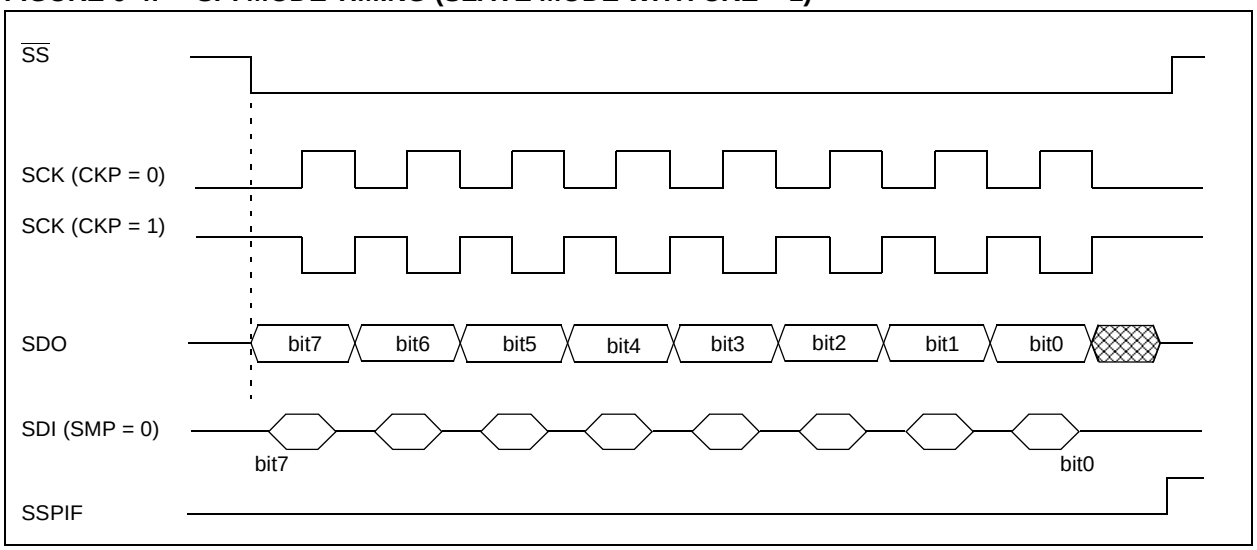


TABLE 9-1 REGISTERS ASSOCIATED WITH SPI OPERATION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR, BOR	$\overline{MCLR}$, WDT
0Bh, 8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
94h	SSPSTAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	0000 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by the SSP in SPI mode.

Note 1: These bits are reserved on the 28-pin devices; always maintain these bits clear.

© 1999 Microchip Technology Inc.

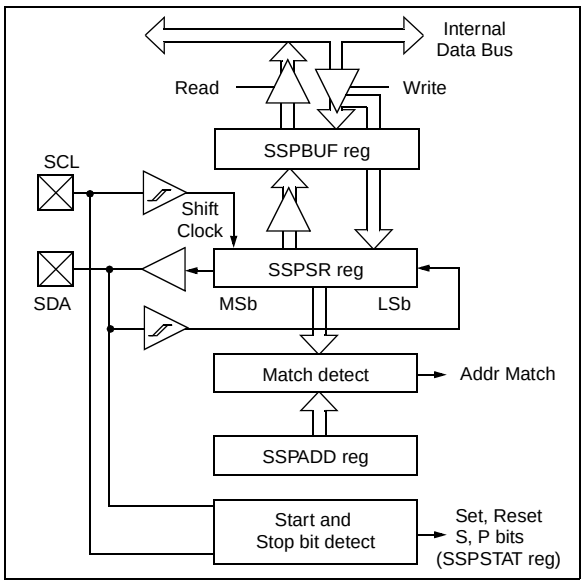
9.2 MSSP I²C Operation

The MSSP module in I²C mode fully implements all master and slave functions (including general call support) and provides interrupts-on-start and stop bits in hardware to determine a free bus (multi-master function). The MSSP module implements the standard mode specifications, as well as 7-bit and 10-bit addressing.

Refer to Application Note AN578, *"Use of the SSP Module in the I²C Multi-Master Environment."*

A "glitch" filter is on the SCL and SDA pins when the pin is an input. This filter operates in both the 100 kHz and 400 kHz modes. In the 100 kHz mode, when these pins are an output, there is a slew rate control of the pin that is independant of device frequency.

FIGURE 9-5: I²C SLAVE MODE BLOCK DIAGRAM



Two pins are used for data transfer. These are the SCL pin, which is the clock, and the SDA pin, which is the data. The SDA and SCL pins are automatically configured when the I²C mode is enabled. The SSP module functions are enabled by setting SSP Enable bit SSPEN (SSPCON<5>).

The MSSP module has six registers for I²C operation. They are the:

- SSP Control Register (SSPCON)
- SSP Control Register2 (SSPCON2)
- SSP Status Register (SSPSTAT)
- Serial Receive/Transmit Buffer (SSPBUF)
- SSP Shift Register (SSPSR) - Not directly accessible
- SSP Address Register (SSPADD)

The SSPCON register allows control of the I²C operation. Four mode selection bits (SSPCON<3:0>) allow one of the following I²C modes to be selected:

- I²C Slave mode (7-bit address)
- I²C Slave mode (10-bit address)
- I²C Master mode, clock = OSC/4 (SSPADD +1)

Before selecting any I²C mode, the SCL and SDA pins must be programmed to inputs by setting the appropriate TRIS bits. Selecting an I²C mode, by setting the SSPEN bit, enables the SCL and SDA pins to be used as the clock and data lines in I²C mode.

The CKE bit (SSPSTAT<6:7>) sets the levels of the SDA and SCL pins in either master or slave mode. When CKE = 1, the levels will conform to the SMBUS specification. When CKE = 0, the levels will conform to the I²C specification.

The SSPSTAT register gives the status of the data transfer. This information includes detection of a START (S) or STOP (P) bit, specifies if the received byte was data or address, if the next byte is the completion of 10-bit address, and if this will be a read or write data transfer.

SSPBUF is the register to which the transfer data is written to or read from. The SSPSR register shifts the data in or out of the device. In receive operations, the SSPBUF and SSPSR create a doubled buffered receiver. This allows reception of the next byte to begin before reading the last byte of received data. When the complete byte is received, it is transferred to the SSPBUF register and flag bit SSPIF is set. If another complete byte is received before the SSPBUF register is read, a receiver overflow has occurred and bit SSPOV (SSPCON<6>) is set and the byte in the SSPSR is lost.

The SSPADD register holds the slave address. In 10-bit mode, the user needs to write the high byte of the address (1111 0 A9 A8 0). Following the high byte address match, the low byte of the address needs to be loaded (A7:A0).

9.2.1 SLAVE MODE

In slave mode, the SCL and SDA pins must be configured as inputs. The MSSP module will override the input state with the output data when required (slave-transmitter).

When an address is matched or the data transfer after an address match is received, the hardware automatically will generate the acknowledge ($\overline{ACK}$) pulse, and then load the SSPBUF register with the received value currently in the SSPSR register.

There are certain conditions that will cause the MSSP module not to give this $\overline{ACK}$ pulse. These are if either (or both):

a) The buffer full bit BF (SSPSTAT<0>) was set before the transfer was received.

b) The overflow bit SSPOV (SSPCON<6>) was set before the transfer was received.

If the BF bit is set, the SSPSR register value is not loaded into the SSPBUF, but bit SSPIF and SSPOV are set. Table 9-2 shows what happens when a data transfer byte is received, given the status of bits BF and SSPOV. The shaded cells show the condition where user software did not properly clear the overflow condition. Flag bit BF is cleared by reading the SSPBUF register, while bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low time for proper operation. The high and low times of the I^2C specification, as well as the requirement of the MSSP module, is shown in timing parameter #100 and parameter #101 of the electrical specifications.

9.2.1.1 ADDRESSING

Once the MSSP module has been enabled, it waits for a START condition to occur. Following the START condition, the 8-bits are shifted into the SSPSR register. All incoming bits are sampled with the rising edge of the clock (SCL) line. The value of register SSPSR<7:1> is compared to the value of the SSPADD register. The address is compared on the falling edge of the eighth clock (SCL) pulse. If the addresses match, and the BF and SSPOV bits are clear, the following events occur:

a) The SSPSR register value is loaded into the SSPBUF register on the falling edge of the 8th SCL pulse.

b) The buffer full bit, BF, is set on the falling edge of the 8th SCL pulse.

c) An $\overline{ACK}$ pulse is generated.

d) SSP interrupt flag bit, SSPIF (PIR1<3>), is set (interrupt is generated if enabled) on the falling edge of the 9th SCL pulse.

In 10-bit address mode, two address bytes need to be received by the slave. The five Most Significant bits (MSbs) of the first address byte specify if this is a 10-bit address. Bit R/$\overline{W}$ (SSPSTAT<2>) must specify a write so the slave device will receive the second address byte. For a 10-bit address the first byte would equal '1111 0 A9 A8 0', where A9 and A8 are the two MSbs of the address. The sequence of events for a 10-bit address is as follows, with steps 7- 9 for slave-transmitter:

1. Receive first (high) byte of Address (bits SSPIF, BF and UA (SSPSTAT<1>) are set).
2. Update the SSPADD register with the second (low) byte of Address (clears bit UA and releases the SCL line).
3. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
4. Receive second (low) byte of Address (bits SSPIF, BF and UA are set).
5. Update the SSPADD register with the first (high) byte of Address. This will clear bit UA and release the SCL line.
6. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
7. Receive Repeated Start condition.
8. Receive first (high) byte of Address (bits SSPIF and BF are set).
9. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.

Note:	Following the Repeated Start condition (step 7) in 10-bit mode, the user only needs to match the first 7-bit address. The user does not update the SSPADD for the second half of the address.

© 1999 Microchip Technology Inc.

9.2.1.2 SLAVE RECEPTION

When the R/$\overline{W}$ bit of the address byte is clear and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is cleared. The received address is loaded into the SSPBUF register.

When the address byte overflow condition exists, then no acknowledge ($\overline{ACK}$) pulse is given. An overflow condition is defined as either bit BF (SSPSTAT<0>) is set or bit SSPOV (SSPCON<6>) is set.

An SSP interrupt is generated for each data transfer byte. Flag bit SSPIF (PIR1<3>) must be cleared in software. The SSPSTAT register is used to determine the status of the received byte.

Note: The SSPBUF will be loaded if the SSPOV bit is set and the BF flag is cleared. If a read of the SSPBUF was performed, but the user did not clear the state of the SSPOV bit before the next receive occurred, the $\overline{ACK}$ is not sent and the SSPBUF is updated.

TABLE 9-2 DATA TRANSFER RECEIVED BYTE ACTIONS

Status Bits as Data Transfer is Received				
BF	**SSPOV**	**SSPSR → SSPBUF**	**Generate $\overline{ACK}$ Pulse**	**Set bit SSPIF (SSP Interrupt occurs if enabled)**
0	0	Yes	Yes	Yes
1	0	No	No	Yes
1	1	No	No	Yes
0	1	Yes	No	Yes

Note 1: Shaded cells show the conditions where the user software did not properly clear the overflow condition.

9.2.1.3 SLAVE TRANSMISSION

When the R/$\overline{W}$ bit of the incoming address byte is set and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is set. The received address is loaded into the SSPBUF register. The $\overline{ACK}$ pulse will be sent on the ninth bit, and the SCL pin is held low. The transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then the SCL pin should be enabled by setting bit CKP (SSPCON<4>). The master must monitor the SCL pin prior to asserting another clock pulse. The slave devices may be holding off the master by stretching the clock. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time (Figure 9-7).

An SSP interrupt is generated for each data transfer byte. The SSPIF flag bit must be cleared in software and the SSPSTAT register is used to determine the status of the byte transfer. The SSPIF flag bit is set on the falling edge of the ninth clock pulse.

As a slave-transmitter, the $\overline{ACK}$ pulse from the master receiver is latched on the rising edge of the ninth SCL input pulse. If the SDA line is high (not $\overline{ACK}$), then the data transfer is complete. When the not $\overline{ACK}$ is latched by the slave, the slave logic is reset and the slave then monitors for another occurrence of the START bit. If the SDA line was low ($\overline{ACK}$), the transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then the SCL pin should be enabled by setting the CKP bit.

FIGURE 9-6: I^2C WAVEFORMS FOR RECEPTION (7-BIT ADDRESS)

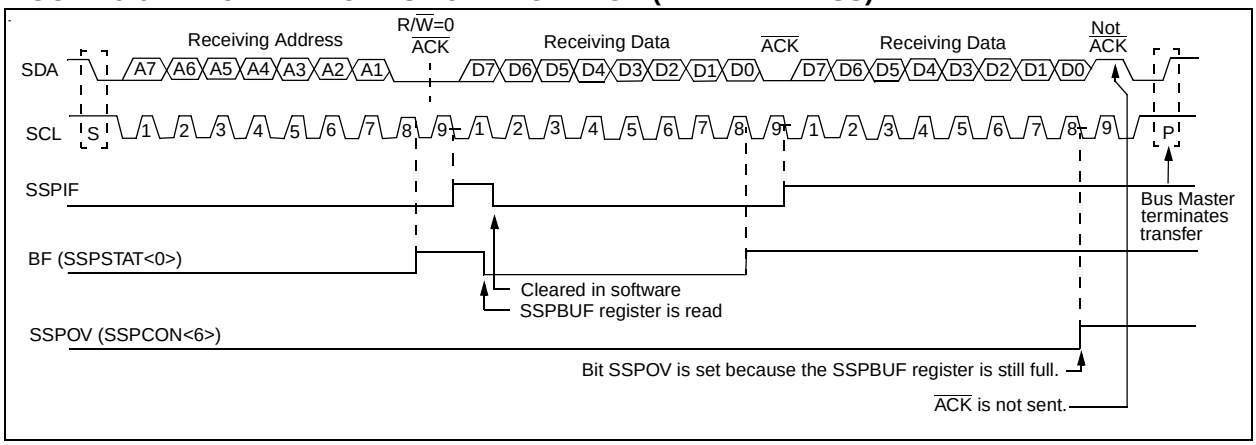
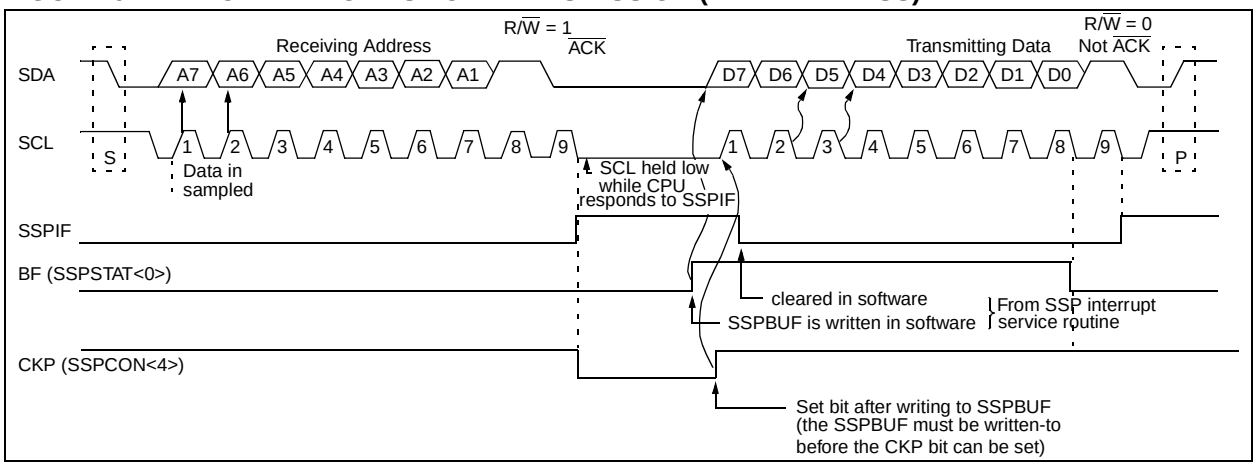


FIGURE 9-7: I²C WAVEFORMS FOR TRANSMISSION (7-BIT ADDRESS)



9.2.2 GENERAL CALL ADDRESS SUPPORT

The addressing procedure for the I²C bus is such that the first byte after the START condition usually determines which device will be the slave addressed by the master. The exception is the general call address, which can address all devices. When this address is used, all devices should, in theory, respond with an acknowledge.

The general call address is one of eight addresses reserved for specific purposes by the I²C protocol. It consists of all 0's with R/$\overline{W}$ = 0

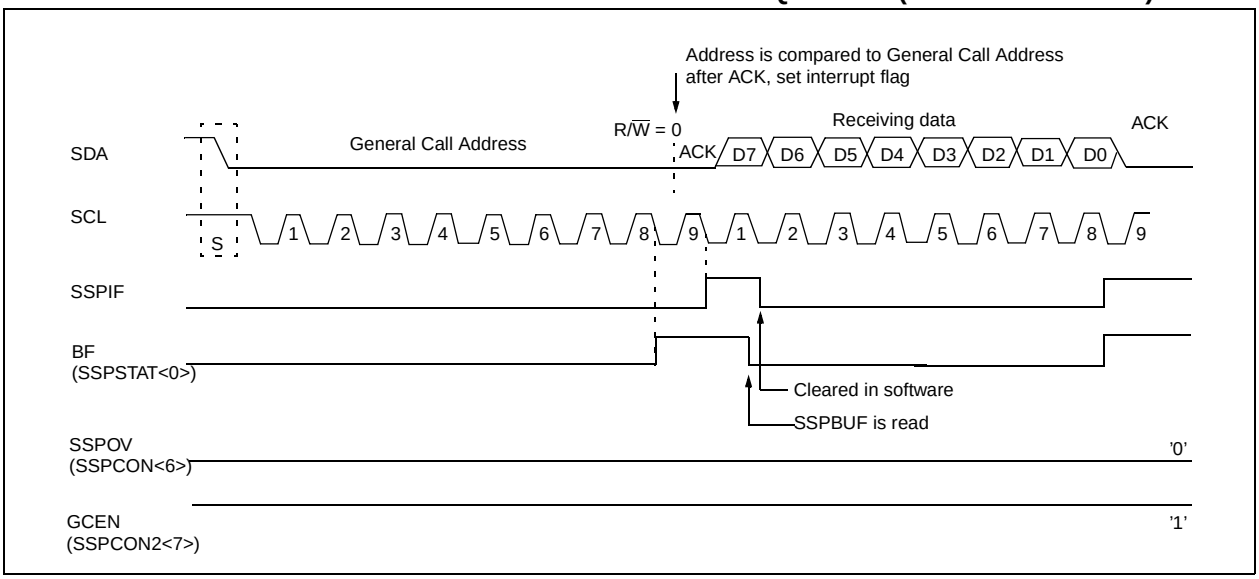
The general call address is recognized when the General Call Enable bit (GCEN) is enabled (SSPCON2<7> is set). Following a start-bit detect, 8-bits are shifted into SSPSR and the address is compared against SSPADD. It is also compared to the general call address and fixed in hardware.

If the general call address matches, the SSPSR is transferred to the SSPBUF, the BF flag is set (eighth bit), and on the falling edge of the ninth bit ($\overline{ACK}$ bit), the SSPIF flag is set.

When the interrupt is serviced, the source for the interrupt can be checked by reading the contents of the SSPBUF to determine if the address was device specific or a general call address.

In 10-bit mode, the SSPADD is required to be updated for the second half of the address to match, and the UA bit is set (SSPSTAT<1>). If the general call address is sampled when GCEN is set while the slave is configured in 10-bit address mode, then the second half of the address is not necessary, the UA bit will not be set, and the slave will begin receiving data after the acknowledge (Figure 9-8).

FIGURE 9-8: SLAVE MODE GENERAL CALL ADDRESS SEQUENCE (7 OR 10-BIT MODE)



© 1999 Microchip Technology Inc.

9.2.3 SLEEP OPERATION

While in sleep mode, the I^2C module can receive addresses or data. When an address match or complete byte transfer occurs, wake the processor from sleep (if the SSP interrupt is enabled).

9.2.4 EFFECTS OF A RESET

A reset disables the SSP module and terminates the current transfer.

TABLE 9-3 REGISTERS ASSOCIATED WITH I^2C OPERATION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR, BOR	$\overline{MCLR}$, WDT
0Bh, 8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
0Dh	PIR2	—	(2)	—	EEIF	BCLIF	—	—	CCP2IF	-r-0 0--0	-r-0 0--0
8Dh	PIE2	—	(2)	—	EEIE	BCLIE	—	—	CCP2IE	-r-0 0--0	-r-0 0--0
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
91h	SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
94h	SSPSTAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	0000 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by the SSP in I^2C mode.

Note 1: These bits are reserved on the 28-pin devices; always maintain these bits clear.

2: These bits are reserved on these devices; always maintain these bits clear.

9.2.5 MASTER MODE

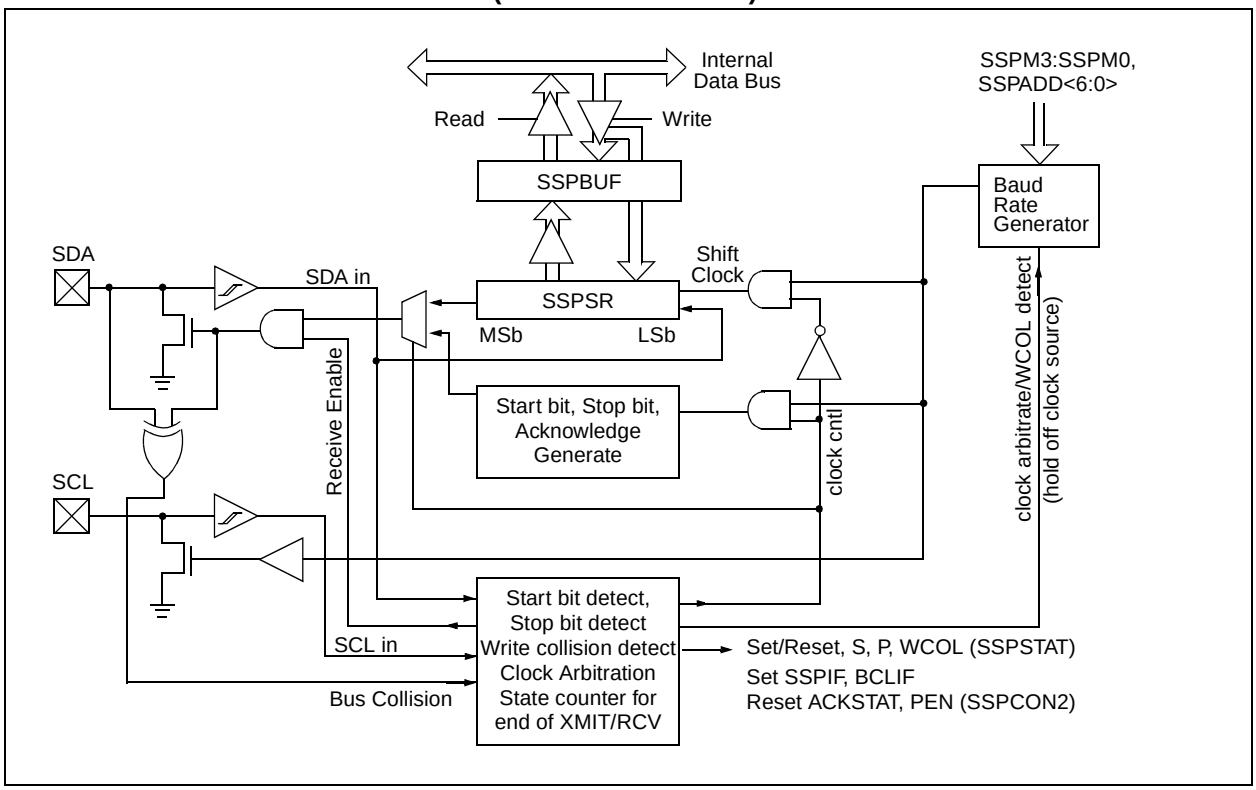
Master mode of operation is supported by interrupt generation on the detection of the START and STOP conditions. The STOP (P) and START (S) bits are cleared from a reset or when the MSSP module is disabled. Control of the I^2C bus may be TACKEN when the P bit is set, or the bus is idle with both the S and P bits clear.

In master mode, the SCL and SDA lines are manipulated by the MSSP hardware.

The following events will cause the SSP Interrupt Flag bit, SSPIF, to be set (SSP Interrupt if enabled):

- START condition
- STOP condition
- Data transfer byte transmitted/received
- Acknowledge transmit
- Repeated Start

FIGURE 9-9: SSP BLOCK DIAGRAM (I^2C MASTER MODE)



© 1999 Microchip Technology Inc.

9.2.6 MULTI-MASTER MODE

In multi-master mode, the interrupt generation on the detection of the START and STOP conditions allows the determination of when the bus is free. The STOP (P) and START (S) bits are cleared from a reset or when the MSSP module is disabled. Control of the I^2C bus may be taken when bit P (SSPSTAT<4>) is set, or the bus is idle with both the S and P bits clear. When the bus is busy, enabling the SSP Interrupt will generate the interrupt when the STOP condition occurs.

In multi-master operation, the SDA line must be monitored for abitration to see if the signal level is the expected output level. This check is performed in hardware, with the result placed in the BCLIF bit.

The states where arbitration can be lost are:

- Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

9.2.7 I^2C MASTER MODE SUPPORT

Master Mode is enabled by setting and clearing the appropriate SSPM bits in SSPCON and by setting the SSPEN bit. Once master mode is enabled, the user has six options.

- Assert a start condition on SDA and SCL.
- Assert a Repeated Start condition on SDA and SCL.
- Write to the SSPBUF register initiating transmission of data/address.
- Generate a stop condition on SDA and SCL.
- Configure the I^2C port to receive data.
- Generate an Acknowledge condition at the end of a received byte of data.

Note: The MSSP Module, when configured in I^2C Master Mode, does not allow queueing of events. For instance, the user is not allowed to initiate a start condition and immediately write the SSPBUF register to initiate transmission before the START condition is complete. In this case, the SSPBUF will not be written to and the WCOL bit will be set, indicating that a write to the SSPBUF did not occur.

9.2.7.1 I^2C MASTER MODE OPERATION

The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a Repeated Start condition. Since the Repeated Start condition is also the beginning of the next serial transfer, the I^2C bus will not be released.

In Master Transmitter mode serial data is output through SDA, while SCL outputs the serial clock. The first byte transmitted contains the slave address of the receiving device (7 bits) and the Read/Write (R/$\overline{W}$) bit. In this case, the R/$\overline{W}$ bit will be logic '0'. Serial data is transmitted 8 bits at a time. After each byte is transmitted, an acknowledge bit is received. START and STOP conditions are output to indicate the beginning and the end of a serial transfer.

In Master receive mode, the first byte transmitted contains the slave address of the transmitting device (7 bits) and the R/$\overline{W}$ bit. In this case, the R/$\overline{W}$ bit will be logic '1'. Thus the first byte transmitted is a 7-bit slave address followed by a '1' to indicate receive bit. Serial data is received via SDA, while SCL outputs the serial clock. Serial data is received 8 bits at a time. After each byte is received, an acknowledge bit is transmitted. START and STOP conditions indicate the beginning and end of transmission.

The baud rate generator used for SPI mode operation is now used to set the SCL clock frequency for either 100 kHz, 400 kHz or 1 MHz I^2C operation. The baud rate generator reload value is contained in the lower 7 bits of the SSPADD register. The baud rate generator will automatically begin counting on a write to the SSPBUF. Once the given operation is complete (i.e. transmission of the last data bit is followed by ACK) the internal clock will automatically stop counting and the SCL pin will remain in its last state

A typical transmit sequence would go as follows:

a) The user generates a Start Condition by setting the START enable bit (SEN) in SSPCON2.
b) SSPIF is set. The module will wait the required start time before any other operation takes place.
c) The user loads the SSPBUF with address to transmit.
d) Address is shifted out the SDA pin until all 8 bits are transmitted.
e) The MSSP Module shifts in the ACK bit from the slave device and writes its value into the SSPCON2 register (SSPCON2<6>).
f) The module generates an interrupt at the end of the ninth clock cycle by setting SSPIF.
g) The user loads the SSPBUF with eight bits of data.
h) DATA is shifted out the SDA pin until all 8 bits are transmitted.

© 1999 Microchip Technology Inc.

i) The MSSP module shifts in the ACK bit from the slave device, and writes its value into the SSPCON2 register (SSPCON2<6>).

j) The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPIF bit.

k) The user generates a STOP condition by setting the STOP enable bit PEN in SSPCON2.

l) Interrupt is generated once the STOP condition is complete.

9.2.8 BAUD RATE GENERATOR

In I^2C master mode, the reload value for the BRG is located in the lower 7 bits of the SSPADD register (Figure 9-10). When the BRG is loaded with this value, the BRG counts down to 0 and stops until another reload has taken place. The BRG count is decremented twice per instruction cycle (TCY), on the Q2 and Q4 clock.

In I^2C master mode, the BRG is reloaded automatically. If Clock Arbitration is taking place for instance, the BRG will be reloaded when the SCL pin is sampled high (Figure 9-11).

FIGURE 9-10: BAUD RATE GENERATOR BLOCK DIAGRAM

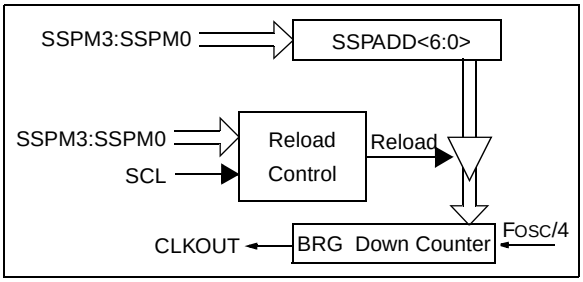
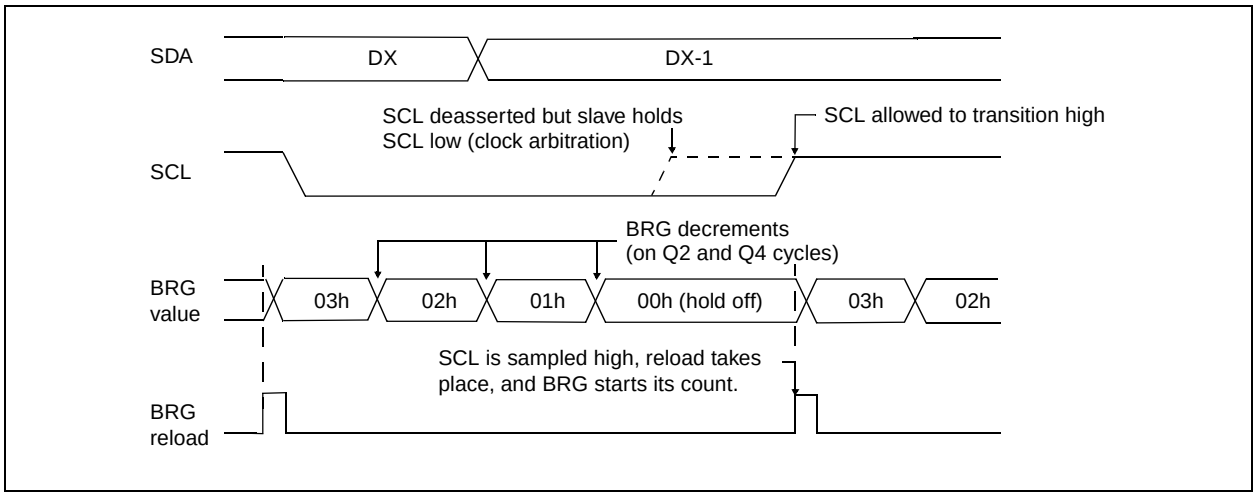


FIGURE 9-11: BAUD RATE GENERATOR TIMING WITH CLOCK ARBITRATION



© 1999 Microchip Technology Inc.

9.2.9 I²C MASTER MODE START CONDITION TIMING

To initiate a START condition, the user sets the start condition enable bit, SEN (SSPCON2<0>). If the SDA and SCL pins are sampled high, the baud rate generator is re-loaded with the contents of SSPADD<6:0> and starts its count. If SCL and SDA are both sampled high when the baud rate generator times out (TBRG), the SDA pin is driven low. The action of the SDA being driven low while SCL is high is the START condition, and causes the S bit (SSPSTAT<3>) to be set. Following this, the baud rate generator is reloaded with the contents of SSPADD<6:0> and resumes its count. When the baud rate generator times out (TBRG), the SEN bit (SSPCON2<0>) will be automatically cleared by hardware. The baud rate generator is suspended leaving the SDA line held low, and the START condition is complete.

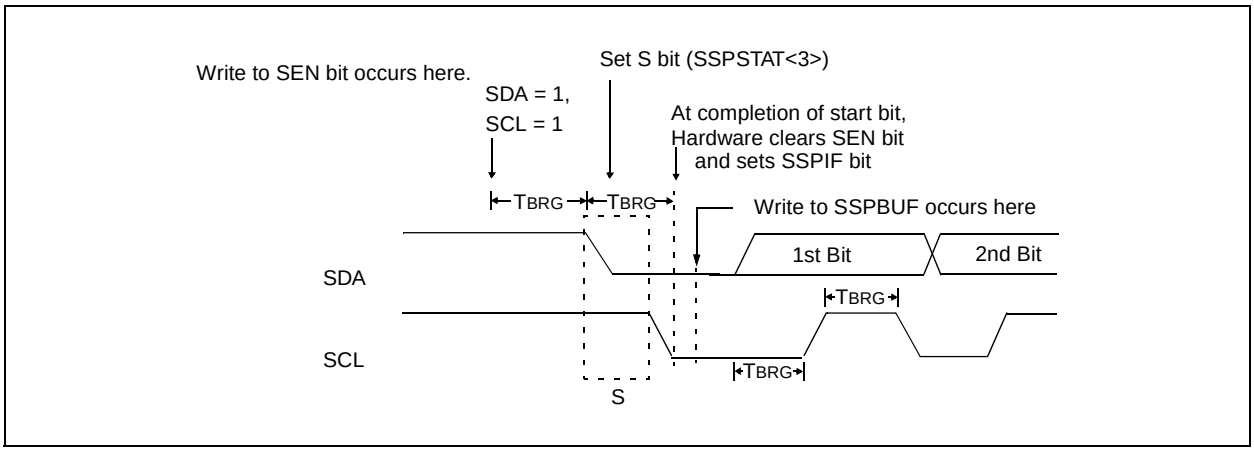
Note:	If at the beginning of START condition the SDA and SCL pins are already sampled low, or if during the START condition the SCL line is sampled low before the SDA line is driven low, a bus collision occurs, the Bus Collision Interrupt Flag (BCLIF) is set, the START condition is aborted, and the I²C module is reset into its IDLE state.

9.2.9.1 WCOL STATUS FLAG

If the user writes the SSPBUF when an START sequence is in progress, then WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

Note:	Because queueing of events is not allowed, writing to the lower 5 bits of SSPCON2 is disabled until the START condition is complete.

FIGURE 9-12: FIRST START BIT TIMING



9.2.10 I²C MASTER MODE REPEATED START CONDITION TIMING

A Repeated Start condition occurs when the RSEN bit (SSPCON2<1>) is programmed high and the I²C module is in the idle state. When the RSEN bit is set, the SCL pin is asserted low. When the SCL pin is sampled low, the baud rate generator is loaded with the contents of SSPADD<6:0> and begins counting. The SDA pin is released (brought high) for one baud rate generator count (TBRG). When the baud rate generator times out if SDA is sampled high, the SCL pin will be deasserted (brought high). When SCL is sampled high the baud rate generator is reloaded with the contents of SSPADD<6:0> and begins counting. SDA and SCL must be sampled high for one TBRG. This action is then followed by assertion of the SDA pin (SDA is low) for one TBRG, while SCL is high. Following this, the RSEN bit in the SSPCON2 register will be automatically cleared and the baud rate generator will not be reloaded, leaving the SDA pin held low. As soon as a start condition is detected on the SDA and SCL pins, the S bit (SSPSTAT<3>) will be set. The SSPIF bit will not be set until the baud rate generator has timed-out.

Note 1: If RSEN is programmed while any other event is in progress, it will not take effect.

Note 2: A bus collision during the Repeated Start condition occurs if:

- SDA is sampled low when SCL goes from low to high.
- SCL goes low before SDA is asserted low. This may indicate that another master is attempting to transmit a data "1".

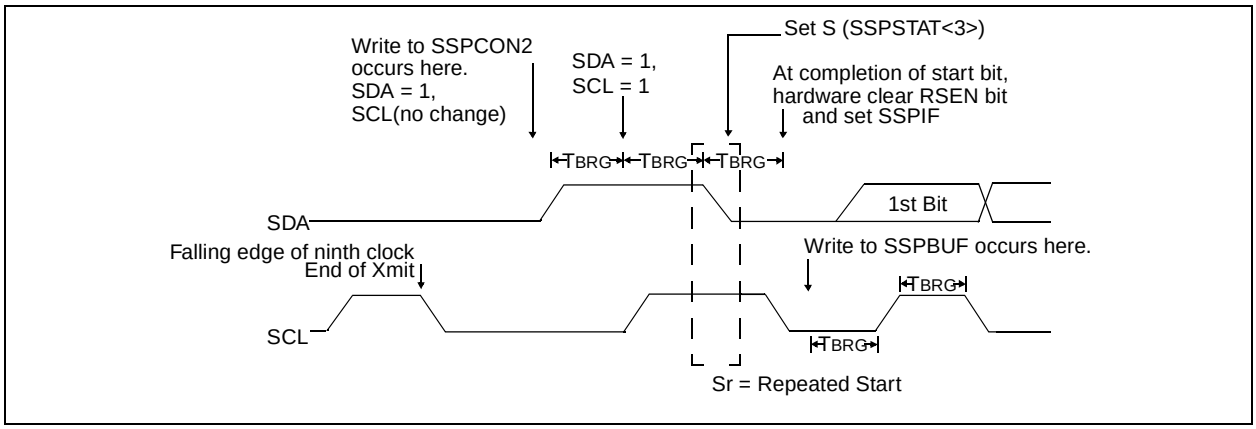
Immediately following the SSPIF bit getting set, the user may write the SSPBUF with the 7-bit address in 7-bit mode, or the default first address in 10-bit mode. After the first eight bits are transmitted and an ACK is received, the user may then transmit an additional eight bits of address (10-bit mode) or eight bits of data (7-bit mode).

9.2.10.1 WCOL STATUS FLAG

If the user writes the SSPBUF when a Repeated Start sequence is in progress, then WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

Note: Because queueing of events is not allowed, writing of the lower 5 bits of SSPCON2 is disabled until the Repeated Start condition is complete.

FIGURE 9-13: REPEAT START CONDITION WAVEFORM



© 1999 Microchip Technology Inc.

9.2.11 I^2C MASTER MODE TRANSMISSION

Transmission of a data byte, a 7-bit address or either half of a 10-bit address is accomplished by simply writing a value to SSPBUF register. This action will set the buffer full flag (BF) and allow the baud rate generator to begin counting and start the next transmission. Each bit of address/data will be shifted out onto the SDA pin after the falling edge of SCL is asserted (see data hold time spec). SCL is held low for one baud rate generator rollover count (TBRG). Data should be valid before SCL is released high (see data setup time spec). When the SCL pin is released high, it is held that way for TBRG. The data on the SDA pin must remain stable for that duration and some hold time after the next falling edge of SCL. After the eighth bit is shifted out (the falling edge of the eighth clock), the BF flag is cleared and the master releases SDA allowing the slave device being addressed to respond with an $\overline{ACK}$ bit during the ninth bit time, if an address match occurs or if data was received properly. The status of $\overline{ACK}$ is read into the ACKDT on the falling edge of the ninth clock. If the master receives an acknowledge, the acknowledge status bit (ACKSTAT) is cleared. If not, the bit is set. After the ninth clock, the SSPIF is set and the master clock (baud rate generator) is suspended until the next data byte is loaded into the SSPBUF, leaving SCL low and SDA unchanged (Figure 9-14).

After the write to the SSPBUF, each bit of address will be shifted out on the falling edge of SCL until all seven address bits and the R/$\overline{W}$ bit are completed. On the falling edge of the eighth clock, the master will de-assert the SDA pin allowing the slave to respond with an acknowledge. On the falling edge of the ninth clock, the master will sample the SDA pin to see if the address was recognized by a slave. The status of the ACK bit is loaded into the ACKSTAT status bit (SSPCON2<6>). Following the falling edge of the ninth clock transmission of the address, the SSPIF is set, the BF flag is cleared, and the baud rate generator is turned off until another write to the SSPBUF takes place, holding SCL low and allowing SDA to float.

9.2.11.1 BF STATUS FLAG

In transmit mode, the BF bit (SSPSTAT<0>) is set when the CPU writes to SSPBUF and is cleared when all 8 bits are shifted out.

9.2.11.2 WCOL STATUS FLAG

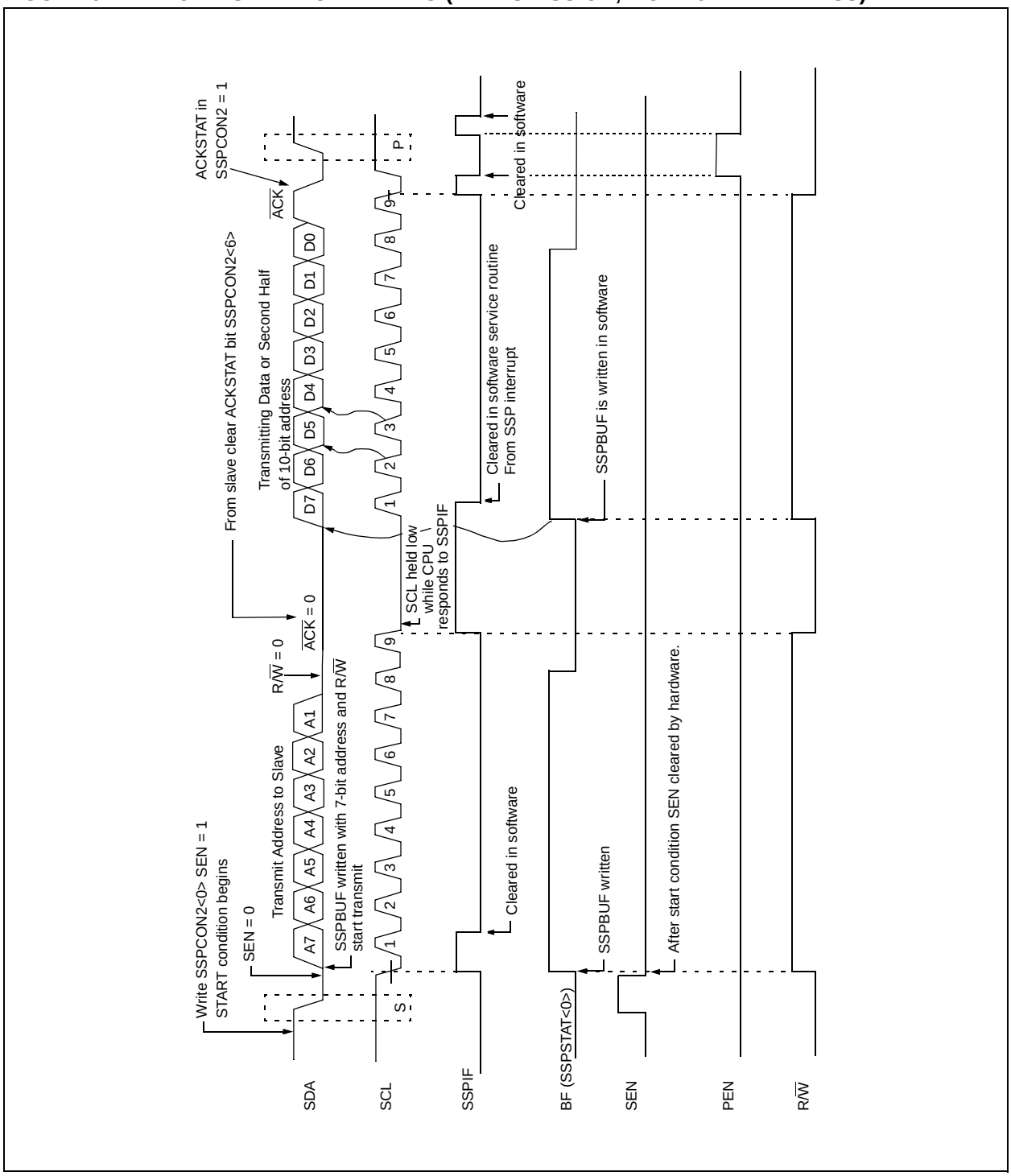
If the user writes the SSPBUF when a transmit is already in progress (i.e. SSPSR is still shifting out a data byte), then WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

WCOL must be cleared in software.

9.2.11.3 ACKSTAT STATUS FLAG

In transmit mode, the ACKSTAT bit (SSPCON2<6>) is cleared when the slave has sent an acknowledge ($\overline{ACK}$ = 0), and is set when the slave does not acknowledge ($\overline{ACK}$ = 1). A slave sends an acknowledge when it has recognized its address (including a general call), or when the slave has properly received its data.

FIGURE 9-14: I²C MASTER MODE TIMING (TRANSMISSION, 7 OR 10-BIT ADDRESS)



© 1999 Microchip Technology Inc.

9.2.12 I^2C MASTER MODE RECEPTION

Master mode reception is enabled by programming the receive enable bit, RCEN (SSPCON2<3>).

Note:	The SSP module must be in an IDLE STATE before the RCEN bit is set or the RCEN bit will be disregarded.

The baud rate generator begins counting, and on each rollover, the state of the SCL pin changes (high to low/low to high), and data is shifted into the SSPSR. After the falling edge of the eighth clock, the receive enable flag is automatically cleared, the contents of the SSPSR are loaded into the SSPBUF, the BF flag is set, the SSPIF is set, and the baud rate generator is suspended from counting, holding SCL low. The SSP is now in IDLE state, awaiting the next command. When the buffer is read by the CPU, the BF flag is automatically cleared. The user can then send an acknowledge bit at the end of reception, by setting the acknowledge sequence enable bit, ACKEN (SSPCON2<4>).

9.2.12.1 BF STATUS FLAG

In receive operation, BF is set when an address or data byte is loaded into SSPBUF from SSPSR. It is cleared when SSPBUF is read.

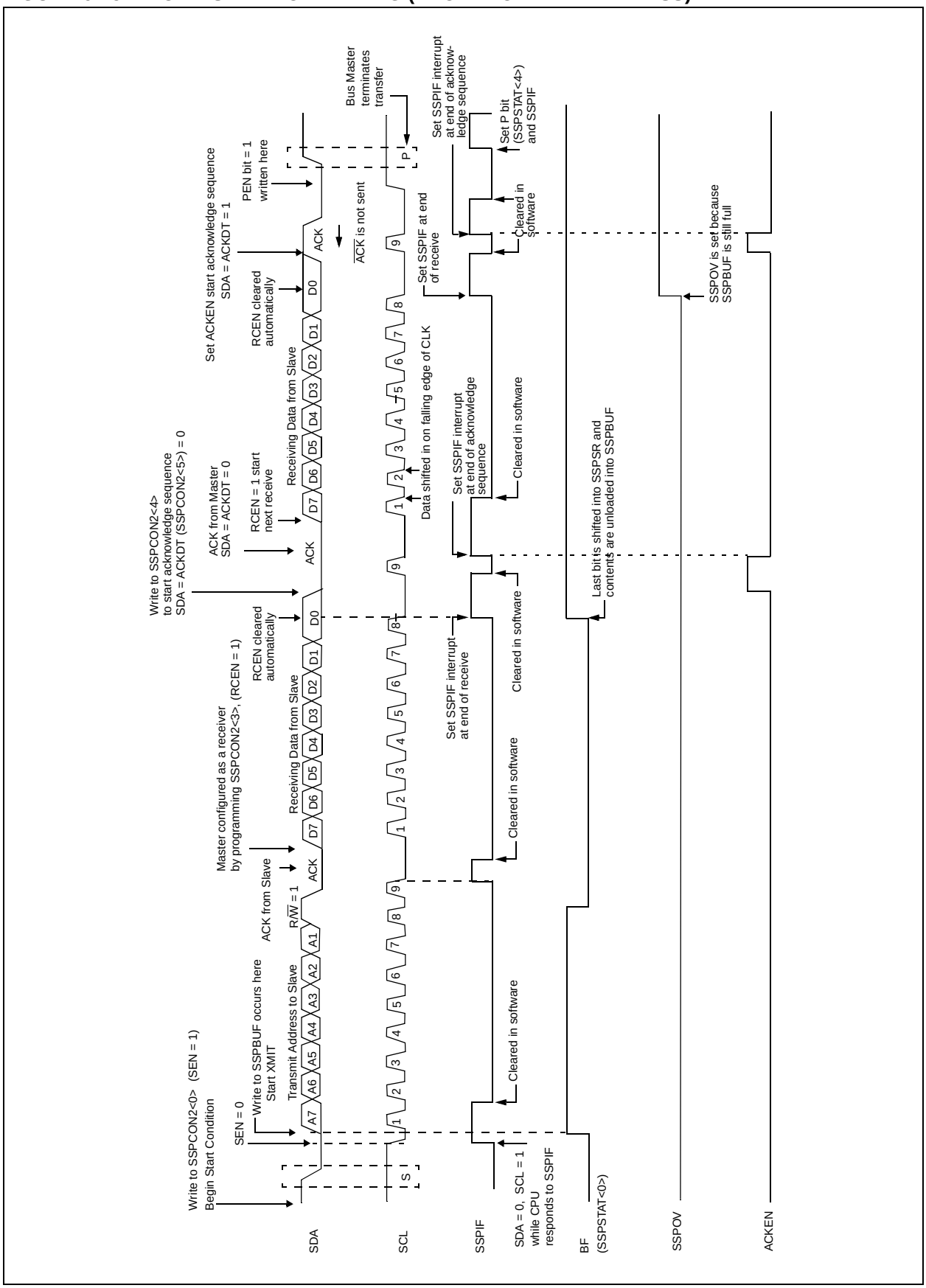
9.2.12.2 SSPOV STATUS FLAG

In receive operation, SSPOV is set when 8 bits are received into the SSPSR, and the BF flag is already set from a previous reception.

9.2.12.3 WCOL STATUS FLAG

If the user writes the SSPBUF when a receive is already in progress (i.e. SSPSR is still shifting in a data byte), then WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

FIGURE 9-15: I²C MASTER MODE TIMING (RECEPTION 7-BIT ADDRESS)



© 1999 Microchip Technology Inc.

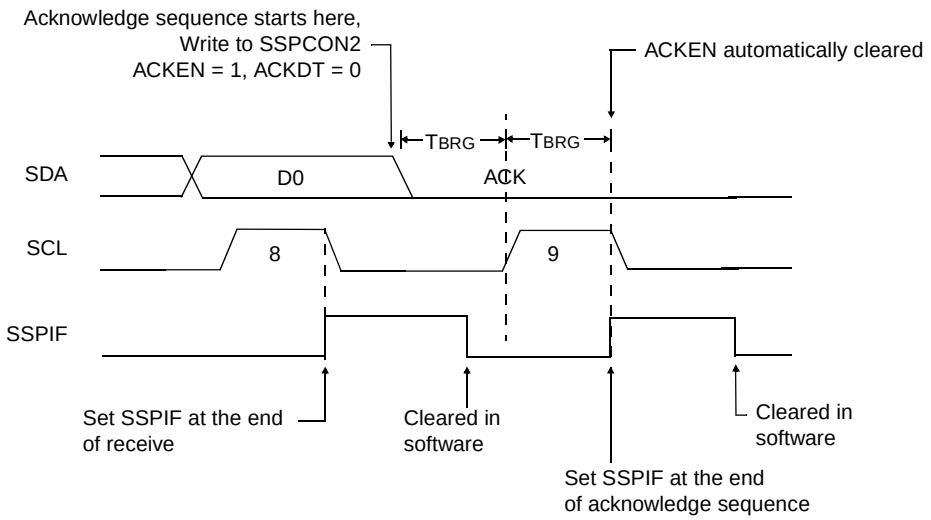
9.2.13 ACKNOWLEDGE SEQUENCE TIMING

An acknowledge sequence is enabled by setting the acknowledge sequence enable bit, ACKEN (SSPCON2<4>). When this bit is set, the SCL pin is pulled low and the contents of the acknowledge data bit is presented on the SDA pin. If the user wishes to generate an acknowledge, the ACKDT bit should be cleared. If not, the user should set the ACKDT bit before starting an acknowledge sequence. The baud rate generator then counts for one rollover period (T_{BRG}), and the SCL pin is deasserted (pulled high). When the SCL pin is sampled high (clock arbitration), the baud rate generator counts for TBRG. The SCL pin is then pulled low. Following this, the ACKEN bit is automatically cleared, the baud rate generator is turned off, and the SSP module then goes into IDLE mode. (Figure 9-16)

9.2.13.1 WCOL STATUS FLAG

If the user writes the SSPBUF when an acknowledege sequence is in progress, the WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

FIGURE 9-16: ACKNOWLEDGE SEQUENCE WAVEFORM



Note: TBRG = one baud rate generator period.

9.2.14 STOP CONDITION TIMING

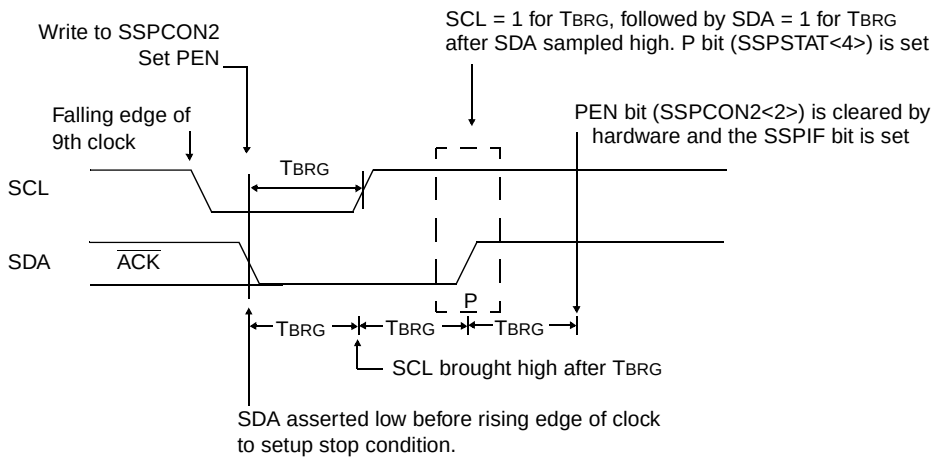
A stop bit is asserted on the SDA pin at the end of a receive/transmit by setting the Stop Sequence Enable bit PEN (SSPCON2<2>). At the end of a receive/transmit, the SCL line is held low after the falling edge of the ninth clock. When the PEN bit is set, the master will assert the SDA line low . When the SDA line is sampled low, the baud rate generator is reloaded and counts down to 0. When the baud rate generator times out, the SCL pin will be brought high, and one TBRG (baud rate generator rollover count) later, the SDA pin will be deasserted. When the SDA pin is sampled high while SCL is high, the P bit (SSPSTAT<4>) is set. A TBRG later, the PEN bit is cleared and the SSPIF bit is set (Figure 9-17).

Whenever the firmware decides to take control of the bus, it will first determine if the bus is busy by checking the S and P bits in the SSPSTAT register. If the bus is busy, then the CPU can be interrupted (notified) when a Stop bit is detected (i.e. bus is free).

9.2.14.1 WCOL STATUS FLAG

If the user writes the SSPBUF when a STOP sequence is in progress, then WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

FIGURE 9-17: STOP CONDITION RECEIVE OR TRANSMIT MODE



Note: TBRG = one baud rate generator period.

© 1999 Microchip Technology Inc.

9.2.15 CLOCK ARBITRATION

Clock arbitration occurs when the master, during any receive, transmit, or repeated start/stop condition, deasserts the SCL pin (SCL allowed to float high). When the SCL pin is allowed to float high, the baud rate generator (BRG) is suspended from counting until the SCL pin is actually sampled high. When the SCL pin is sampled high, the baud rate generator is reloaded with the contents of SSPADD<6:0> and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count in the event that the clock is held low by an external device (Figure 9-18).

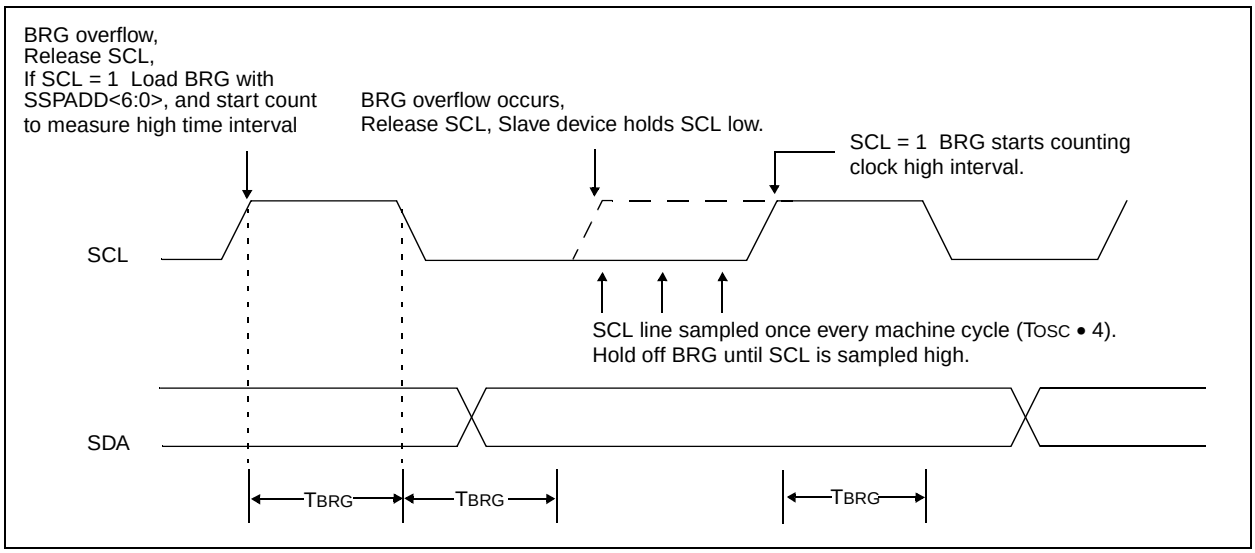
9.2.16 SLEEP OPERATION

While in sleep mode, the I^2C module can receive addresses or data, and when an address match or complete byte transfer occurs, wake the processor from sleep (if the SSP interrupt is enabled).

9.2.17 EFFECTS OF A RESET

A reset disables the SSP module and terminates the current transfer.

FIGURE 9-18: CLOCK ARBITRATION TIMING IN MASTER TRANSMIT MODE



9.2.18 MULTI -MASTER COMMUNICATION, BUS COLLISION, AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin = '0', a bus collision has TACKEN place. The master will set the Bus Collision Interrupt Flag, BCLIF and reset the I^2C port to its IDLE state. (Figure 9-19).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are deasserted, and the SSPBUF can be written to. When the user services the bus collision interrupt service routine, and if the I^2C bus is free, the user can resume communication by asserting a START condition.

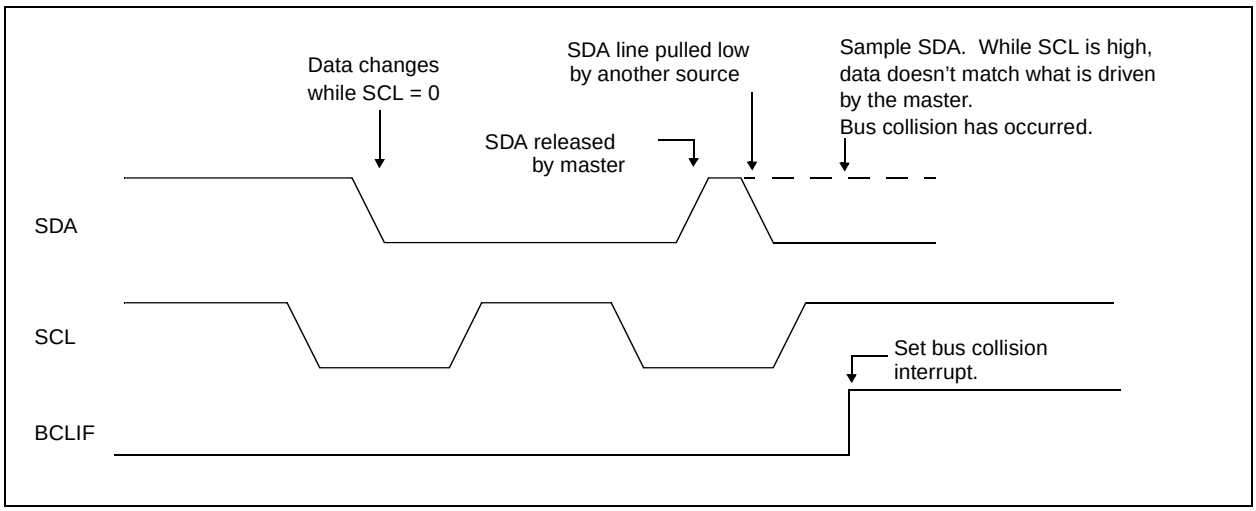
If a START, Repeated Start, STOP or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted, and the respective control bits in the SSPCON2 register are cleared. When the user services the bus collision interrupt service routine, and if the I^2C bus is free, the user can resume communication by asserting a START condition.

The Master will continue to monitor the SDA and SCL pins, and if a STOP condition occurs, the SSPIF bit will be set.

A write to the SSPBUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when the bus collision occurred.

In multi-master mode, the interrupt generation on the detection of start and stop conditions allows the determination of when the bus is free. Control of the I^2C bus can be TACKEN when the P bit is set in the SSPSTAT register, or the bus is idle and the S and P bits are cleared.

FIGURE 9-19: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



© 1999 Microchip Technology Inc.

9.2.18.1 BUS COLLISION DURING A START CONDITION

During a START condition, a bus collision occurs if:

a) SDA or SCL are sampled low at the beginning of the START condition (Figure 9-20).

b) SCL is sampled low before SDA is asserted low. (Figure 9-21).

During a START condition both the SDA and the SCL pins are monitored.

If:

the SDA pin is already low
or the SCL pin is already low,

then:

the START condition is aborted,
and the BCLIF flag is set,
and the SSP module is reset to its IDLE state (Figure 9-20).

The START condition begins with the SDA and SCL pins deasserted. When the SDA pin is sampled high, the baud rate generator is loaded from SSPADD<6:0> and counts down to 0. If the SCL pin is sampled low while SDA is high, a bus collision occurs, because it is assumed that another master is attempting to drive a data '1' during the START condition.

If the SDA pin is sampled low during this count, the BRG is reset and the SDA line is asserted early (Figure 9-22). If however a '1' is sampled on the SDA pin, the SDA pin is asserted low at the end of the BRG count. The baud rate generator is then reloaded and counts down to 0. During this time, if the SCL pins are sampled as '0', a bus collision does not occur. At the end of the BRG count ,the SCL pin is asserted low.

Note: The reason that bus collision is not a factor during a START condition is that no two bus masters can assert a START condition at the exact same time. Therefore, one master will always assert SDA before the other. This condition does not cause a bus collision, because the two masters must be allowed to arbitrate the first address following the START condition. If the address is the same, arbitration must be allowed to continue into the data portion, REPEATED START or STOP conditions.

FIGURE 9-20: BUS COLLISION DURING START CONDITION (SDA ONLY)

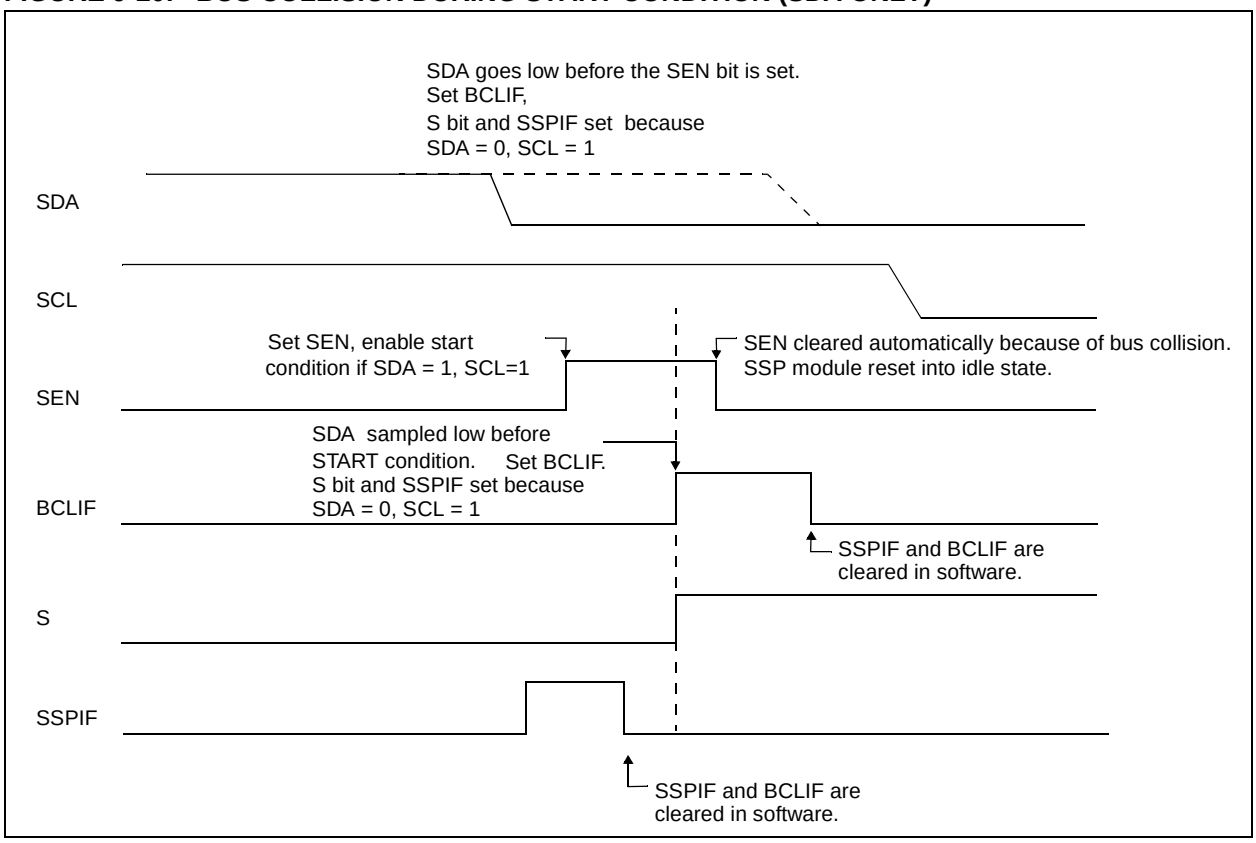


FIGURE 9-21: BUS COLLISION DURING START CONDITION (SCL = 0)

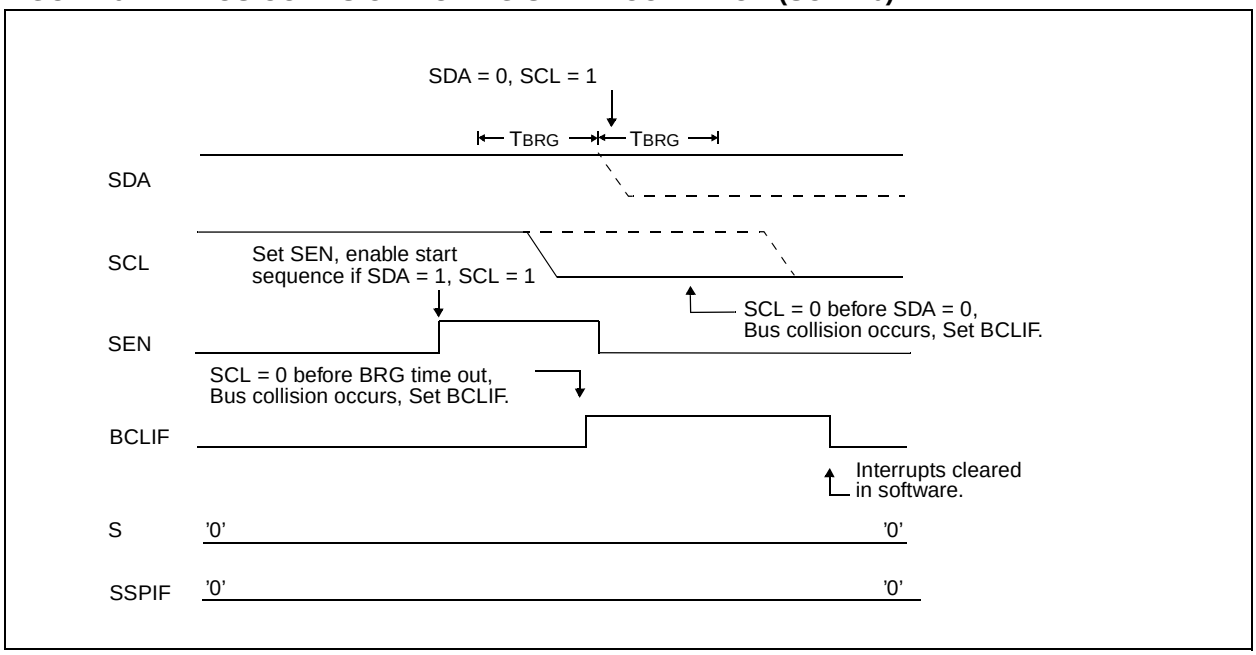
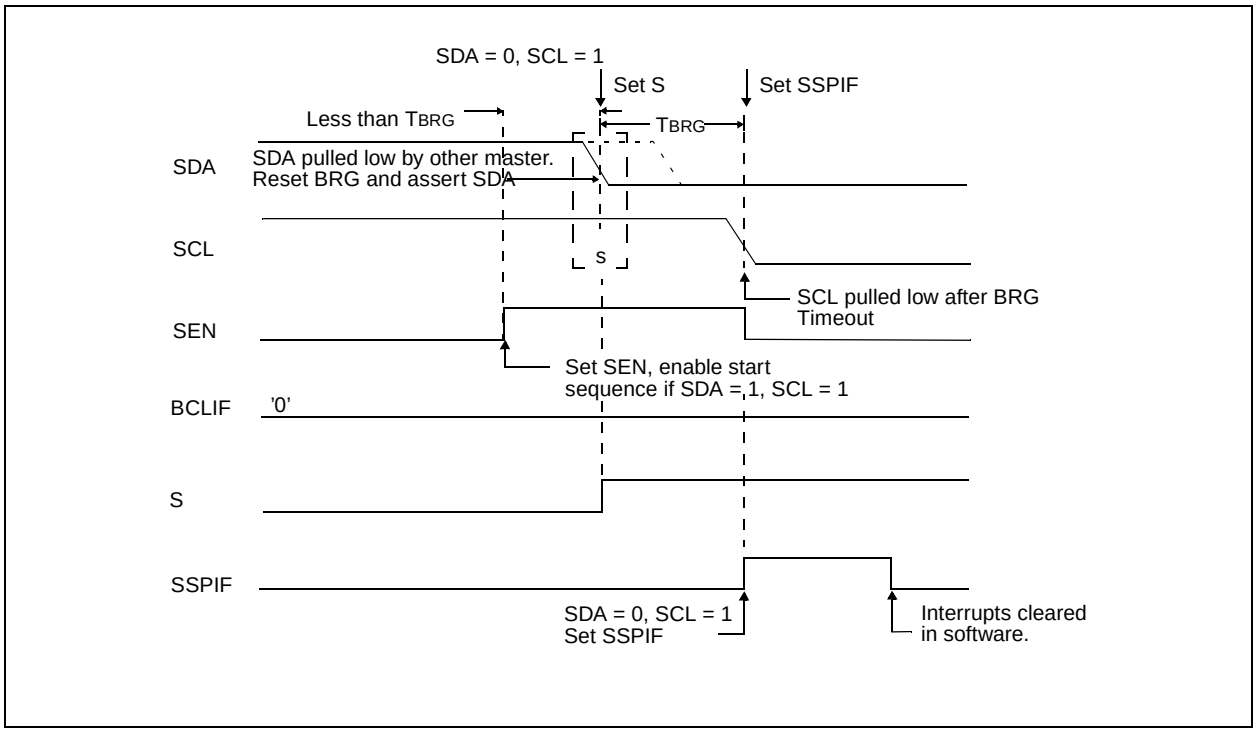


FIGURE 9-22: BRG RESET DUE TO SDA COLLISION DURING START CONDITION



© 1999 Microchip Technology Inc.

9.2.18.2 BUS COLLISION DURING A REPEATED START CONDITION

During a Repeated Start condition, a bus collision occurs if:

a) A low level is sampled on SDA when SCL goes from low level to high level.

b) SCL goes low before SDA is asserted low, indicating that another master is attempting to transmit a data '1'.

When the user deasserts SDA and the pin is allowed to float high, the BRG is loaded with SSPADD<6:0> and counts down to 0. The SCL pin is then deasserted, and when sampled high, the SDA pin is sampled. If SDA is low, a bus collision has occurred (i.e. another master is attempting to transmit a data '0'). If however SDA is sampled high, the BRG is reloaded and begins counting. If SDA goes from high to low before the BRG times out, no bus collision occurs, because no two masters can assert SDA at exactly the same time.

If, however, SCL goes from high to low before the BRG times out and SDA has not already been asserted, a bus collision occurs. In this case, another master is attempting to transmit a data '1' during the Repeated Start condition.

If at the end of the BRG time out both SCL and SDA are still high, the SDA pin is driven low, the BRG is reloaded and begins counting. At the end of the count, regardless of the status of the SCL pin, the SCL pin is driven low and the Repeated Start condition is complete (Figure 9-23).

FIGURE 9-23: BUS COLLISION DURING A REPEATED START CONDITION (CASE 1)

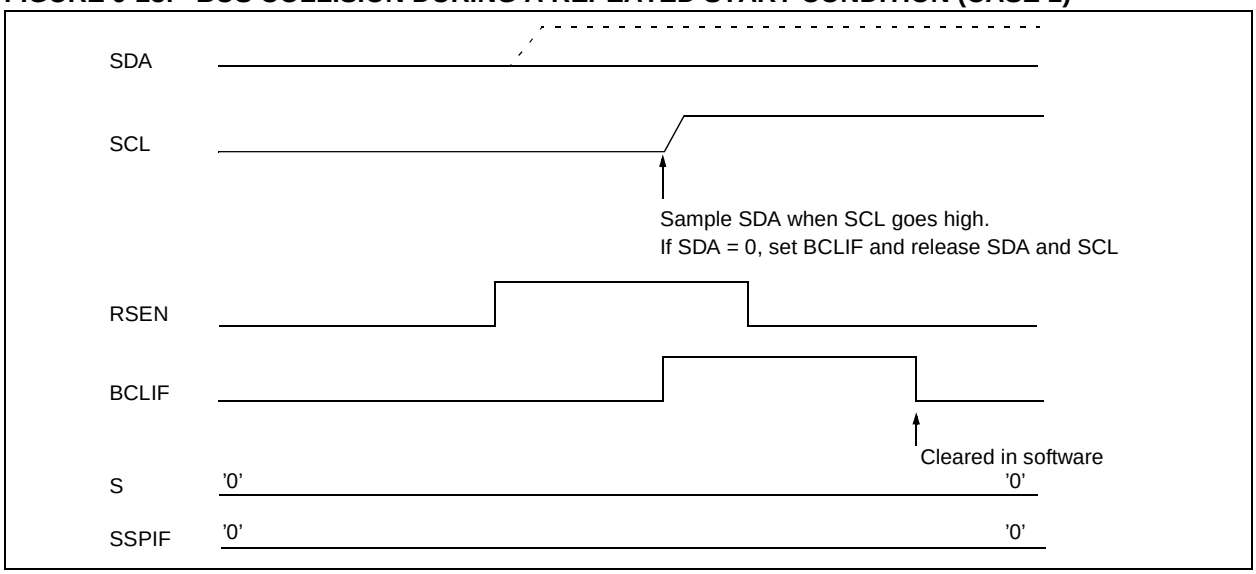
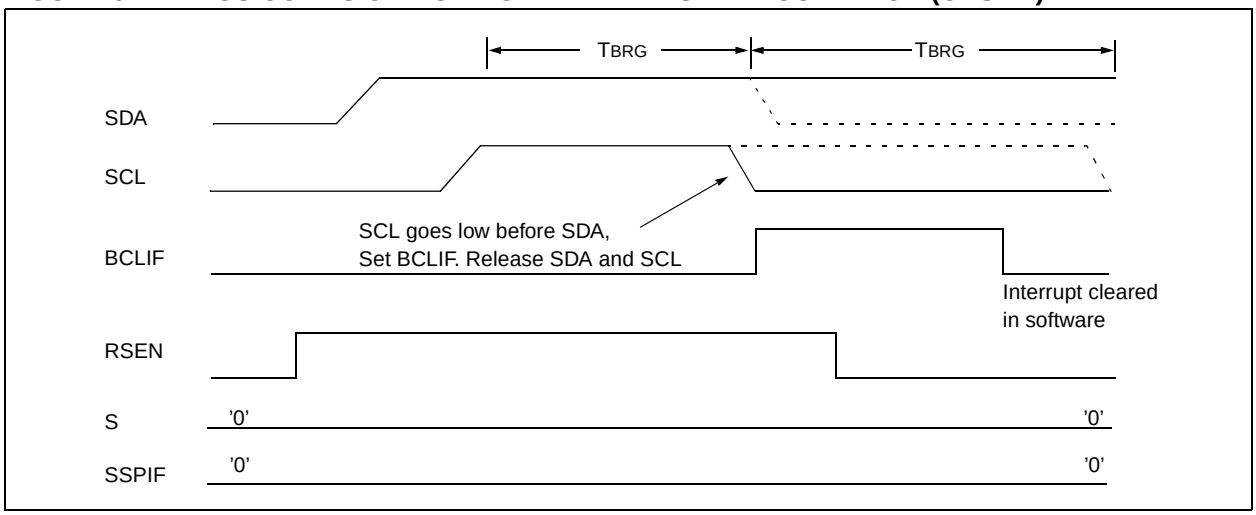


FIGURE 9-24: BUS COLLISION DURING REPEATED START CONDITION (CASE 2)



9.2.18.3 BUS COLLISION DURING A STOP CONDITION

Bus collision occurs during a STOP condition if:

a) After the SDA pin has been deasserted and allowed to float high, SDA is sampled low after the BRG has timed out.

b) After the SCL pin is deasserted, SCL is sampled low before SDA goes high.

The STOP condition begins with SDA asserted low. When SDA is sampled low, the SCL pin is allow to float. When the pin is sampled high (clock arbitration), the baud rate generator is loaded with SSPADD<6:0> and counts down to 0. After the BRG times out, SDA is sampled. If SDA is sampled low, a bus collision has occurred. This is due to another master attempting to drive a data '0'. If the SCL pin is sampled low before SDA is allowed to float high, a bus collision occurs. This is a case of another master attempting to drive a data '0' (Figure 9-25).

FIGURE 9-25: BUS COLLISION DURING A STOP CONDITION (CASE 1)

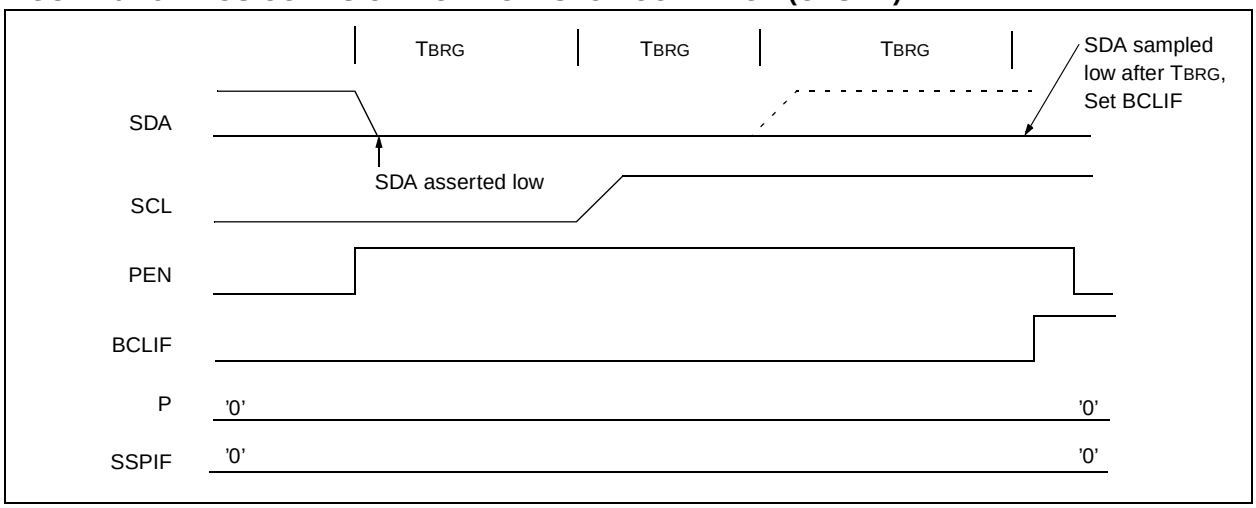
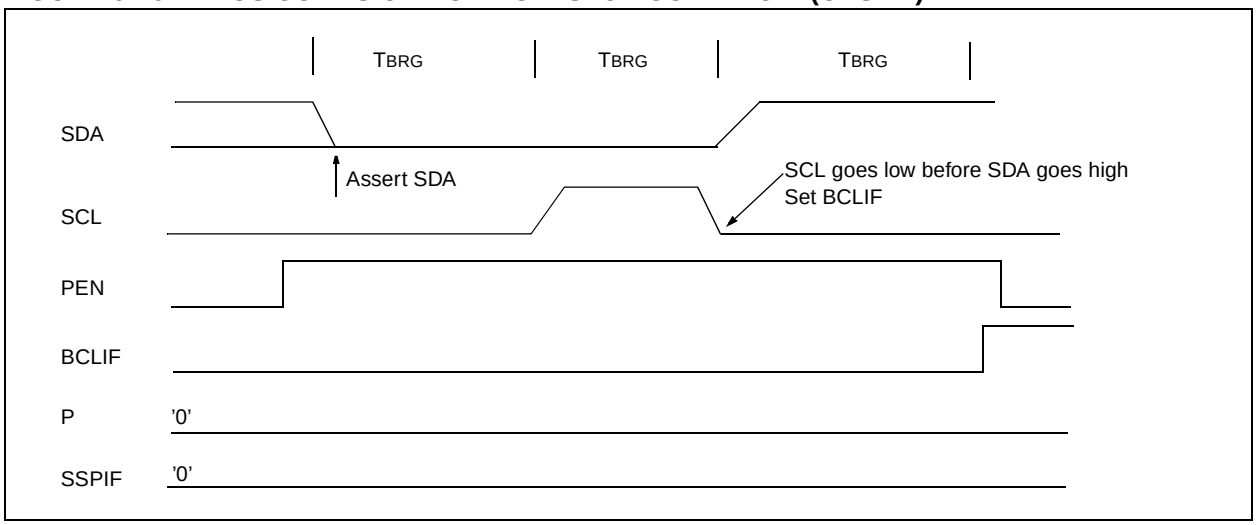


FIGURE 9-26: BUS COLLISION DURING A STOP CONDITION (CASE 2)



© 1999 Microchip Technology Inc.

9.3 Connection Considerations for I²C Bus

For standard-mode I²C bus devices, the values of resistors R_p and R_s in Figure 9-27 depend on the following parameters:

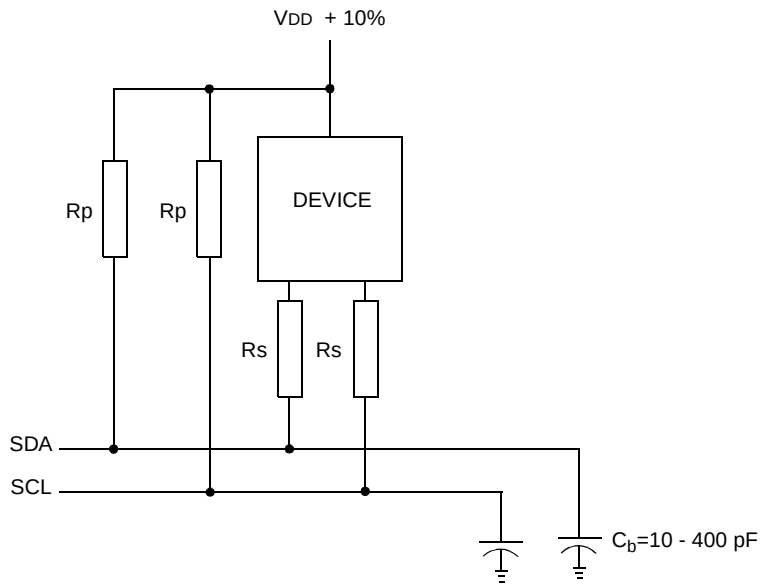
- Supply voltage
- Bus capacitance
- Number of connected devices (input current + leakage current).

The supply voltage limits the minimum value of resistor R_p due to the specified minimum sink current of 3 mA at VOL max = 0.4V for the specified output stages. For example, with a supply voltage of VDD = 5V±10% and VOL max = 0.4V at 3 mA, $R_{p\ min}$ = (5.5-0.4)/0.003 = 1.7 kΩ. VDD as a function of R_p is shown in Figure 9-27. The desired noise margin of 0.1VDD for the low level limits the maximum value of R_s. Series resistors are optional and used to improve ESD susceptibility.

The bus capacitance is the total capacitance of wire, connections, and pins. This capacitance limits the maximum value of R_p due to the specified rise time (Figure 9-27).

The SMP bit is the slew rate control enabled bit. This bit is in the SSPSTAT register, and controls the slew rate of the I/O pins when in I²C mode (master or slave).

FIGURE 9-27: SAMPLE DEVICE CONFIGURATION FOR I²C BUS



Note: I²C devices with input levels related to VDD must have one common supply line to which the pull-up resistor is also connected.

NOTES:

© 1999 Microchip Technology Inc.

10.0 ADDRESSABLE UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (USART)

The Universal Synchronous Asynchronous Receiver Transmitter (USART) module is one of the two serial I/O modules. (USART is also known as a Serial Communications Interface or SCI). The USART can be configured as a full duplex asynchronous system that can communicate with peripheral devices such as CRT terminals and personal computers, or it can be configured as a half duplex synchronous system that can communicate with peripheral devices such as A/D or D/A integrated circuits, serial EEPROMs etc.

The USART can be configured in the following modes:

- Asynchronous (full duplex)
- Synchronous - Master (half duplex)
- Synchronous - Slave (half duplex)

Bit SPEN (RCSTA<7>) and bits TRISC<7:6> have to be set in order to configure pins RC6/TX/CK and RC7/RX/DT as the Universal Synchronous Asynchronous Receiver Transmitter.

The USART module also has a multi-processor communication capability using 9-bit address detection.

REGISTER 10-1: TXSTA: TRANSMIT STATUS AND CONTROL REGISTER (ADDRESS 98h)

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **CSRC:** Clock Source Select bit
Asynchronous mode
Don't care
Synchronous mode
1 = Master mode (Clock generated internally from BRG)
0 = Slave mode (Clock from external source)

bit 6: **TX9**: 9-bit Transmit Enable bit
1 = Selects 9-bit transmission
0 = Selects 8-bit transmission

bit 5: **TXEN**: Transmit Enable bit
1 = Transmit enabled
0 = Transmit disabled
Note: SREN/CREN overrides TXEN in SYNC mode.

bit 4: **SYNC**: USART Mode Select bit
1 = Synchronous mode
0 = Asynchronous mode

bit 3: **Unimplemented:** Read as '0'

bit 2: **BRGH**: High Baud Rate Select bit
Asynchronous mode
1 = High speed
0 = Low speed
Synchronous mode
Unused in this mode

bit 1: **TRMT**: Transmit Shift Register Status bit
1 = TSR empty
0 = TSR full

bit 0: **TX9D:** 9th bit of transmit data. Can be parity bit.

REGISTER 10-2: RCSTA: RECEIVE STATUS AND CONTROL REGISTER (ADDRESS 18h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	R-0	R-x
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **SPEN:** Serial Port Enable bit
1 = Serial port enabled (Configures RC7/RX/DT and RC6/TX/CK pins as serial port pins)
0 = Serial port disabled

bit 6: **RX9**: 9-bit Receive Enable bit
1 = Selects 9-bit reception
0 = Selects 8-bit reception

bit 5: **SREN**: Single Receive Enable bit
Asynchronous mode
Don't care
Synchronous mode - master
1 = Enables single receive
0 = Disables single receive
This bit is cleared after reception is complete.
Synchronous mode - slave
Unused in this mode

bit 4: **CREN**: Continuous Receive Enable bit
Asynchronous mode
1 = Enables continuous receive
0 = Disables continuous receive
Synchronous mode
1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN)
0 = Disables continuous receive

bit 3: **ADDEN:** Address Detect Enable bit
Asynchronous mode 9-bit (RX9 = 1)
1 = Enables address detection, enable interrupt and load of the receive burffer when RSR<8> is set
0 = Disables address detection, all bytes are received, and ninth bit can be used as parity bit

bit 2: **FERR**: Framing Error bit
1 = Framing error (Can be updated by reading RCREG register and receive next valid byte)
0 = No framing error

bit 1: **OERR**: Overrun Error bit
1 = Overrun error (Can be cleared by clearing bit CREN)
0 = No overrun error

bit 0: **RX9D:** 9th bit of received data (Can be parity bit)

© 1999 Microchip Technology Inc.

10.1 USART Baud Rate Generator (BRG)

The BRG supports both the asynchronous and synchronous modes of the USART. It is a dedicated 8-bit baud rate generator. The SPBRG register controls the period of a free running 8-bit timer. In asynchronous mode, bit BRGH (TXSTA<2>) also controls the baud rate. In synchronous mode, bit BRGH is ignored. Table 10-1 shows the formula for computation of the baud rate for different USART modes which only apply in master mode (internal clock).

Given the desired baud rate and Fosc, the nearest integer value for the SPBRG register can be calculated using the formula in Table 10-1. From this, the error in baud rate can be determined.

It may be advantageous to use the high baud rate (BRGH = 1) even for slower baud clocks. This is because the FOSC/(16(X + 1)) equation can reduce the baud rate error in some cases.

Writing a new value to the SPBRG register causes the BRG timer to be reset (or cleared). This ensures the BRG does not wait for a timer overflow before outputting the new baud rate.

10.1.1 SAMPLING

The data on the RC7/RX/DT pin is sampled three times by a majority detect circuit to determine if a high or a low level is present at the RX pin.

TABLE 10-1: BAUD RATE FORMULA

SYNC	BRGH = 0 (Low Speed)	BRGH = 1 (High Speed)
0	(Asynchronous) Baud Rate = FOSC/(64(X+1))	Baud Rate= FOSC/(16(X+1))
1	(Synchronous) Baud Rate = FOSC/(4(X+1))	NA

X = value in SPBRG (0 to 255)

TABLE 10-2: REGISTERS ASSOCIATED WITH BAUD RATE GENERATOR

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used by the BRG.

TABLE 10-3: BAUD RATES FOR ASYNCHRONOUS MODE (BRGH = 0)

BAUD RATE (K)	Fosc = 20 MHz			Fosc = 16 MHz			Fosc = 10 MHz		
	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)
0.3	-	-	-	-	-	-	-	-	-
1.2	1.221	1.75	255	1.202	0.17	207	1.202	0.17	129
2.4	2.404	0.17	129	2.404	0.17	103	2.404	0.17	64
9.6	9.766	1.73	31	9.615	0.16	25	9.766	1.73	15
19.2	19.531	1.72	15	19.231	0.16	12	19.531	1.72	7
28.8	31.250	8.51	9	27.778	3.55	8	31.250	8.51	4
33.6	34.722	3.34	8	35.714	6.29	6	31.250	6.99	4
57.6	62.500	8.51	4	62.500	8.51	3	52.083	9.58	2
HIGH	1.221	-	255	0.977	-	255	0.610	-	255
LOW	312.500	-	0	250.000	-	0	156.250	-	0

BAUD RATE (K)	Fosc = 4 MHz			Fosc = 3.6864 MHz		
	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)
0.3	0.300	0	207	0.301	0.33	185
1.2	1.202	0.17	51	1.216	1.33	46
2.4	2.404	0.17	25	2.432	1.33	22
9.6	8.929	6.99	6	9.322	2.90	5
19.2	20.833	8.51	2	18.643	2.90	2
28.8	31.250	8.51	1	-	-	-
33.6	-	-	-	-	-	-
57.6	62.500	8.51	0	55.930	2.90	0
HIGH	0.244	-	255	0.218	-	255
LOW	62.500	-	0	55.930	-	0

-

TABLE 10-4: BAUD RATES FOR ASYNCHRONOUS MODE (BRGH = 1)

BAUD RATE (K)	Fosc = 20 MHz			Fosc = 16 MHz			Fosc = 10 MHz		
	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)
0.3	-	-	-	-	-	-	-	-	-
1.2	-	-	-	-	-	-	-	-	-
2.4	-	-	-	-	-	-	2.441	1.71	255
9.6	9.615	0.16	129	9.615	0.16	103	9.615	0.16	64
19.2	19.231	0.16	64	19.231	0.16	51	19.531	1.72	31
28.8	29.070	0.94	42	29.412	2.13	33	28.409	1.36	21
33.6	33.784	0.55	36	33.333	0.79	29	32.895	2.10	18
57.6	59.524	3.34	20	58.824	2.13	16	56.818	1.36	10
HIGH	4.883	-	255	3.906	-	255	2.441	-	255
LOW	1250.000	-	0	1000.000		0	625.000	-	0

BAUD RATE (K)	Fosc = 4 MHz			Fosc = 3.6864 MHz		
	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)
0.3	-	-	-	-	-	-
1.2	1.202	0.17	207	1.203	0.25	185
2.4	2.404	0.17	103	2.406	0.25	92
9.6	9.615	0.16	25	9.727	1.32	22
19.2	19.231	0.16	12	18.643	2.90	11
28.8	27.798	3.55	8	27.965	2.90	7
33.6	35.714	6.29	6	31.960	4.88	6
57.6	62.500	8.51	3	55.930	2.90	3
HIGH	0.977	-	255	0.874	-	255
LOW	250.000	-	0	273.722	-	0

© 1999 Microchip Technology Inc.

10.2 USART Asynchronous Mode

In this mode, the USART uses standard non-return-to-zero (NRZ) format (one start bit, eight or nine data bits, and one stop bit). The most common data format is 8 bits. An on-chip, dedicated, 8-bit baud rate generator can be used to derive standard baud rate frequencies from the oscillator. The USART transmits and receives the LSb first. The USART's transmitter and receiver are functionally independent, but use the same data format and baud rate. The baud rate generator produces a clock either x16 or x64 of the bit shift rate, depending on bit BRGH (TXSTA<2>). Parity is not supported by the hardware, but can be implemented in software (and stored as the ninth data bit). Asynchronous mode is stopped during SLEEP.

Asynchronous mode is selected by clearing bit SYNC (TXSTA<4>).

The USART Asynchronous module consists of the following important elements:

- Baud Rate Generator
- Sampling Circuit
- Asynchronous Transmitter
- Asynchronous Receiver

10.2.1 USART ASYNCHRONOUS TRANSMITTER

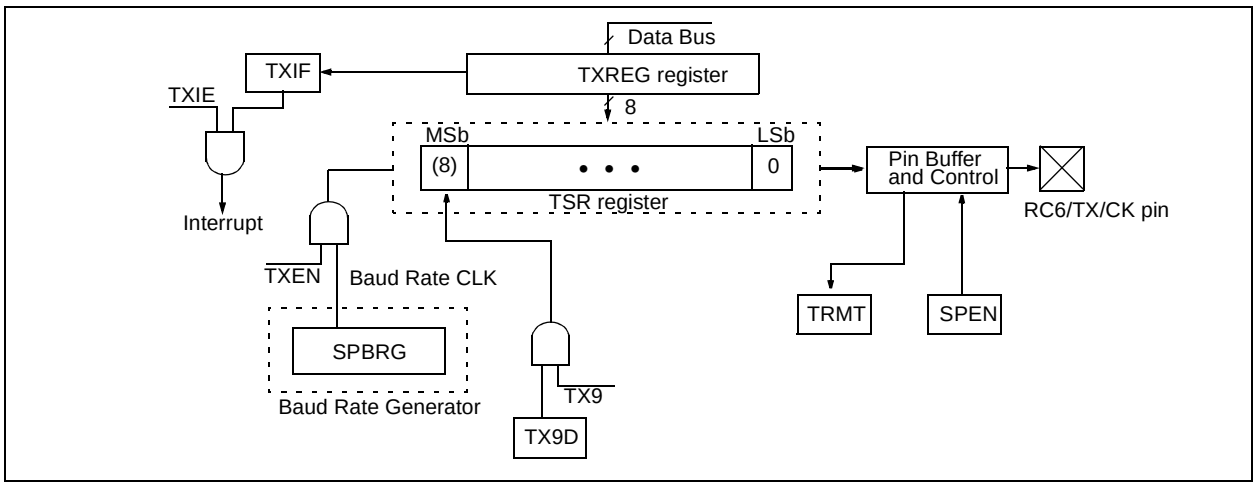
The USART transmitter block diagram is shown in Figure 10-1. The heart of the transmitter is the transmit (serial) shift register (TSR). The shift register obtains its data from the read/write transmit buffer, TXREG. The TXREG register is loaded with data in software. The TSR register is not loaded until the STOP bit has been transmitted from the previous load. As soon as the STOP bit is transmitted, the TSR is loaded with new data from the TXREG register (if available). Once the TXREG register transfers the data to the TSR register (occurs in one TCY), the TXREG register is empty and flag bit TXIF (PIR1<4>) is set. This interrupt can be enabled/disabled by setting/clearing enable bit TXIE (PIE1<4>). Flag bit TXIF will be set, regardless of the state of enable bit TXIE and cannot be cleared in software. It will reset only when new data is loaded into the TXREG register. While flag bit TXIF indicates the status of the TXREG register, another bit TRMT (TXSTA<1>) shows the status of the TSR register. Status bit TRMT is a read only bit, which is set when the TSR register is empty. No interrupt logic is tied to this bit, so the user has to poll this bit in order to determine if the TSR register is empty.

Note 1:	The TSR register is not mapped in data memory, so it is not available to the user.
2:	Flag bit TXIF is set when enable bit TXEN is set. TXIF is cleared by loading TXREG.

Transmission is enabled by setting enable bit TXEN (TXSTA<5>). The actual transmission will not occur until the TXREG register has been loaded with data and the baud rate generator (BRG) has produced a shift clock (Figure 10-2). The transmission can also be started by first loading the TXREG register and then setting enable bit TXEN. Normally, when transmission is first started, the TSR register is empty. At that point, transfer to the TXREG register will result in an immediate transfer to TSR, resulting in an empty TXREG. A back-to-back transfer is thus possible (Figure 10-3). Clearing enable bit TXEN during a transmission will cause the transmission to be aborted and will reset the transmitter. As a result, the RC6/TX/CK pin will revert to hi-impedance.

In order to select 9-bit transmission, transmit bit TX9 (TXSTA<6>) should be set and the ninth bit should be written to TX9D (TXSTA<0>). The ninth bit must be written before writing the 8-bit data to the TXREG register. This is because a data write to the TXREG register can result in an immediate transfer of the data to the TSR register (if the TSR is empty). In such a case, an incorrect ninth data bit may be loaded in the TSR register.

FIGURE 10-1: USART TRANSMIT BLOCK DIAGRAM



Steps to follow when setting up an Asynchronous Transmission:

1. Initialize the SPBRG register for the appropriate baud rate. If a high speed baud rate is desired, set bit BRGH. (Section 10.1)
2. Enable the asynchronous serial port by clearing bit SYNC and setting bit SPEN.
3. If interrupts are desired, then set enable bit TXIE.
4. If 9-bit transmission is desired, then set transmit bit TX9.
5. Enable the transmission by setting bit TXEN, which will also set bit TXIF.
6. If 9-bit transmission is selected, the ninth bit should be loaded in bit TX9D.
7. Load data to the TXREG register (starts transmission).

FIGURE 10-2: ASYNCHRONOUS MASTER TRANSMISSION

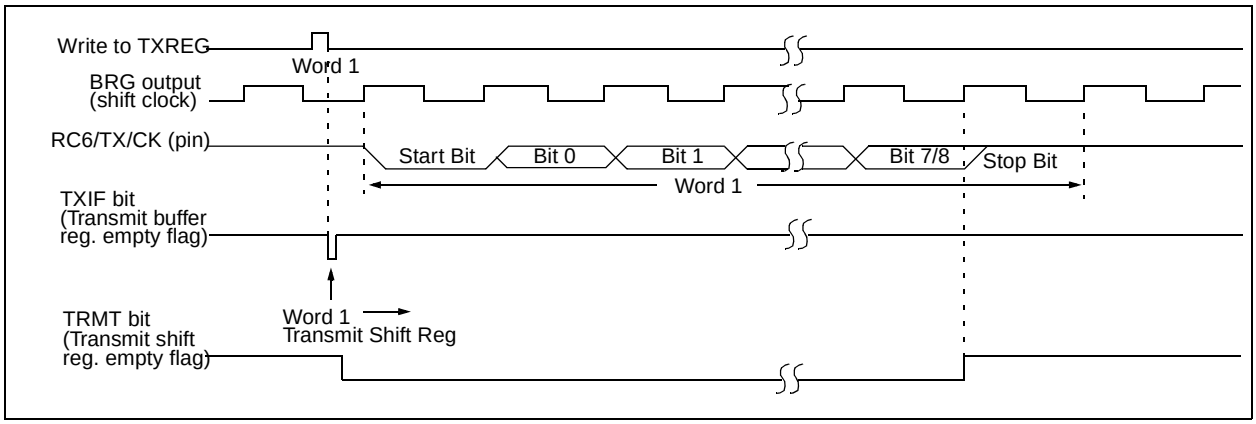
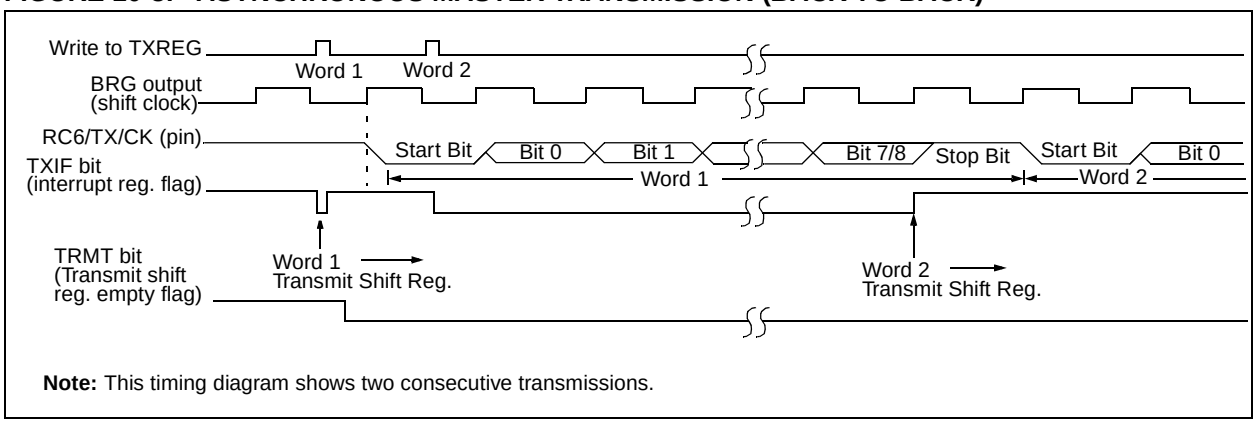


FIGURE 10-3: ASYNCHRONOUS MASTER TRANSMISSION (BACK TO BACK)



Note: This timing diagram shows two consecutive transmissions.

TABLE 10-5: REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	USART Transmit Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for asynchronous transmission.
Note 1: Bits PSPIE and PSPIF are reserved on the PIC16F873/876; always maintain these bits clear.

© 1999 Microchip Technology Inc.

10.2.2 USART ASYNCHRONOUS RECEIVER

The receiver block diagram is shown in Figure 10-4. The data is received on the RC7/RX/DT pin and drives the data recovery block. The data recovery block is actually a high speed shifter operating at x16 times the baud rate, whereas the main receive serial shifter operates at the bit rate or at FOSC.

Once asynchronous mode is selected, reception is enabled by setting bit CREN (RCSTA<4>).

The heart of the receiver is the receive (serial) shift register (RSR). After sampling the STOP bit, the received data in the RSR is transferred to the RCREG register (if it is empty). If the transfer is complete, flag bit RCIF (PIR1<5>) is set. The actual interrupt can be enabled/disabled by setting/clearing enable bit RCIE (PIE1<5>). Flag bit RCIF is a read only bit which is cleared by the hardware. It is cleared when the RCREG register has been read and is empty. The RCREG is a double buffered register (i.e. it is a two deep FIFO). It is possible for two bytes of data to be received and transferred to the RCREG FIFO and a third byte to begin shifting to the RSR register. On the detection of the STOP bit of the third byte, if the RCREG register is still full, the overrun error bit OERR (RCSTA<1>) will be set. The word in the RSR will be lost. The RCREG register can be read twice to retrieve the two bytes in the FIFO. Overrun bit OERR has to be cleared in software. This is done by resetting the receive logic (CREN is cleared and then set). If bit OERR is set, transfers from the RSR register to the RCREG register are inhibited, so it is essential to clear error bit OERR if it is set. Framing error bit FERR (RCSTA<2>) is set if a stop bit is detected as clear. Bit FERR and the 9th receive bit are buffered the same way as the receive data. Reading the RCREG will load bits RX9D and FERR with new values, therefore it is essential for the user to read the RCSTA register before reading RCREG register in order not to lose the old FERR and RX9D information.

FIGURE 10-4: USART RECEVE BLOCK DIAGRAM

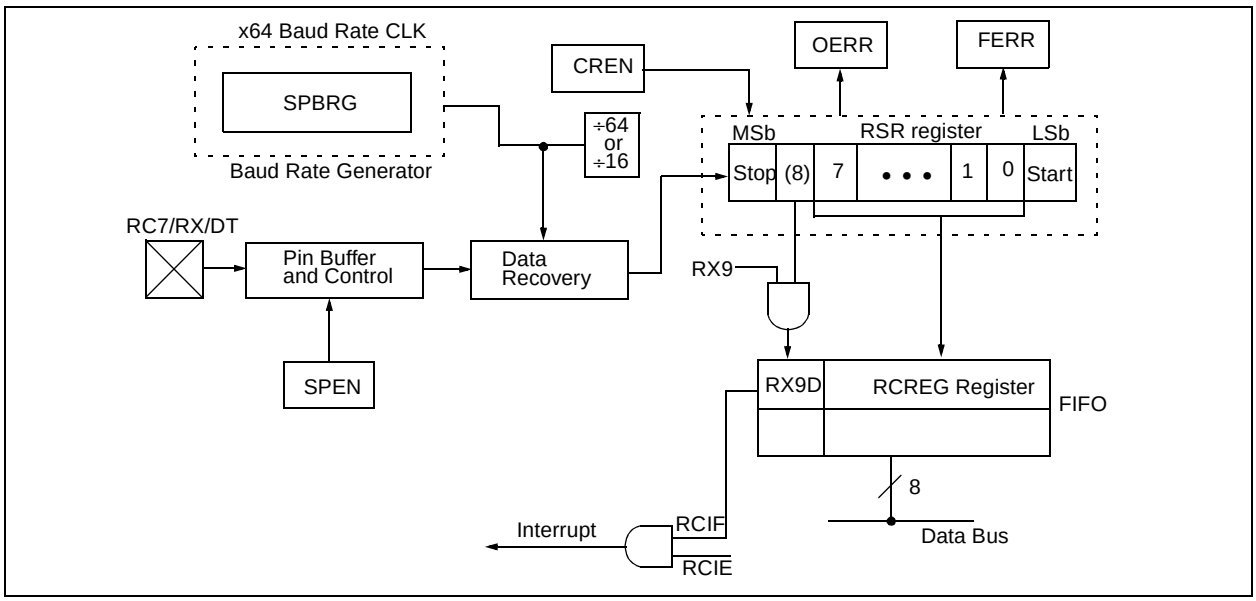
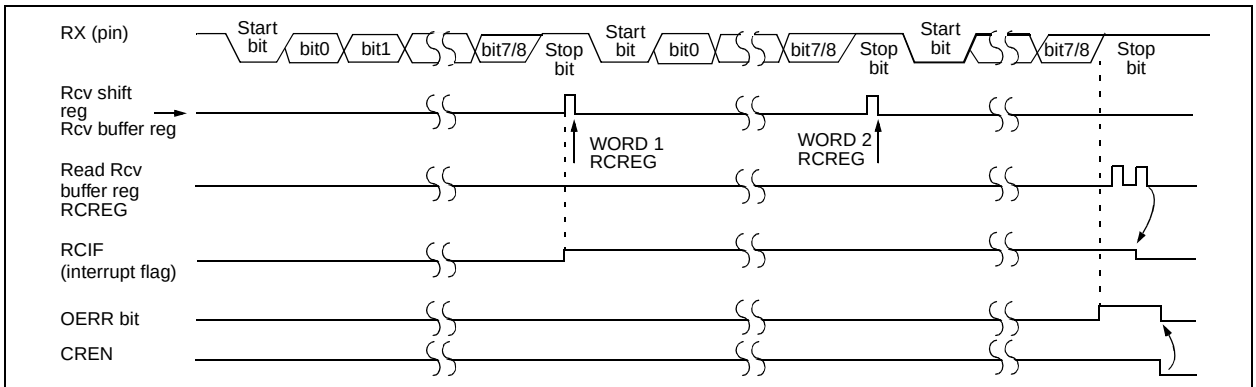


FIGURE 10-5: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RX input. The RCREG (receive buffer) is read after the third word, causing the OERR (overrun) bit to be set.

Steps to follow when setting up an Asynchronous Reception:

1. Initialize the SPBRG register for the appropriate baud rate. If a high speed baud rate is desired, set bit BRGH. (Section 10.1).
2. Enable the asynchronous serial port by clearing bit SYNC and setting bit SPEN.
3. If interrupts are desired, then set enable bit RCIE.
4. If 9-bit reception is desired, then set bit RX9.
5. Enable the reception by setting bit CREN.
6. Flag bit RCIF will be set when reception is complete and an interrupt will be generated if enable bit RCIE is set.
7. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
8. Read the 8-bit received data by reading the RCREG register.
9. If any error occurred, clear the error by clearing enable bit CREN.

TABLE 10-6: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for asynchronous reception.
Note 1: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

© 1999 Microchip Technology Inc.

10.2.3 SETTING UP 9-BIT MODE WITH ADDRESS DETECT

Steps to follow when setting up an Asynchronous Reception with Address Detect Enabled:

- Initialize the SPBRG register for the appropriate baud rate. If a high speed baud rate is desired, set bit BRGH.
- Enable the asynchronous serial port by clearing bit SYNC and setting bit SPEN.
- If interrupts are desired, then set enable bit RCIE.
- Set bit RX9 to enable 9-bit reception.
- Set ADDEN to enable address detect.
- Enable the reception by setting enable bit CREN.
- Flag bit RCIF will be set when reception is complete, and an interrupt will be generated if enable bit RCIE was set.
- Read the RCSTA register to get the ninth bit and determine if any error occurred during reception.
- Read the 8-bit received data by reading the RCREG register, to determine if the device is being addressed.
- If any error occurred, clear the error by clearing enable bit CREN.
- If the device has been addressed, clear the ADDEN bit to allow data bytes and address bytes to be read into the receive buffer, and interrupt the CPU.

FIGURE 10-6: USART RECEIVE BLOCK DIAGRAM

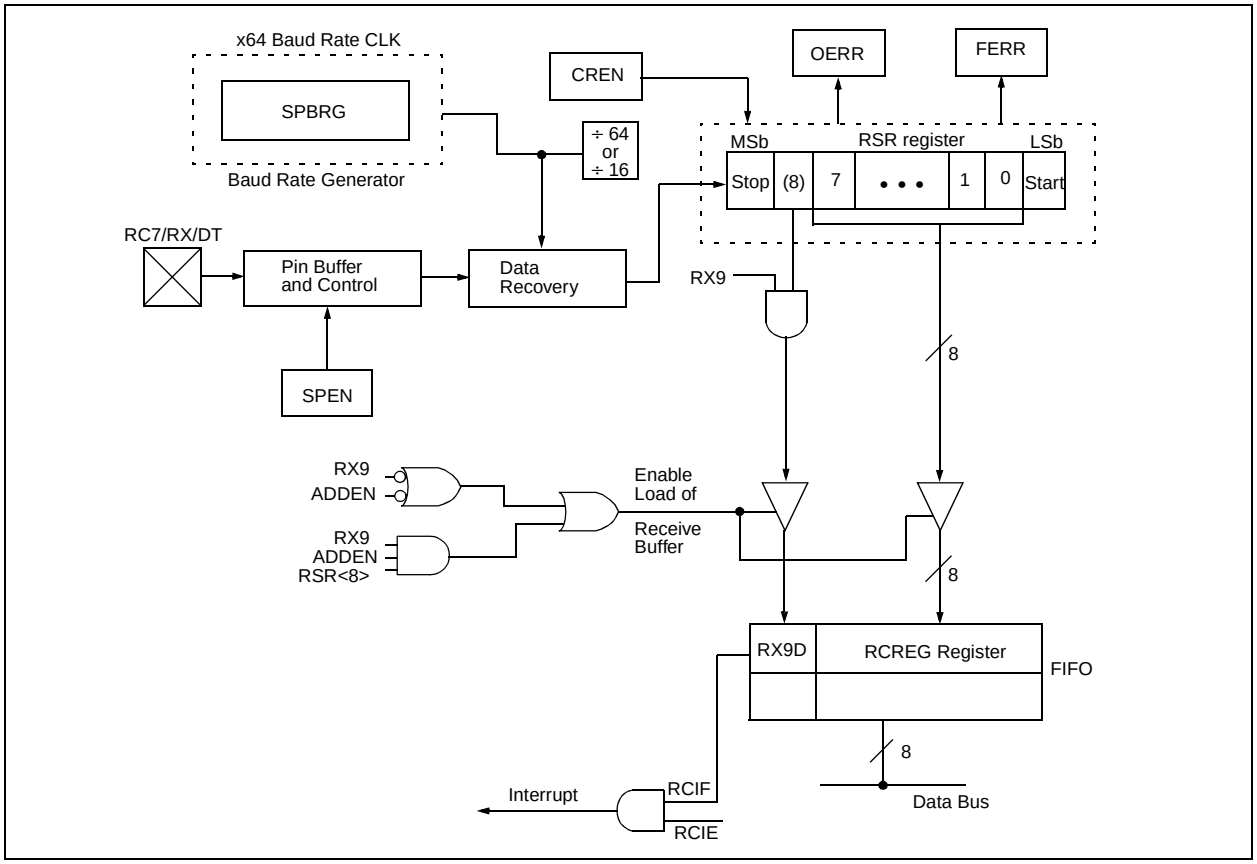
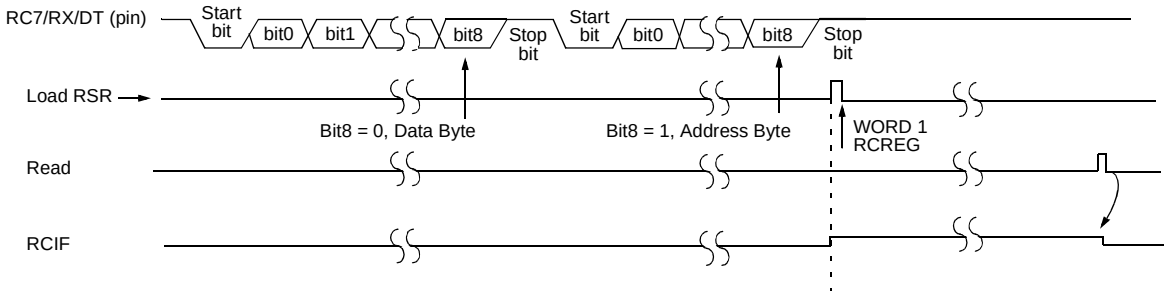
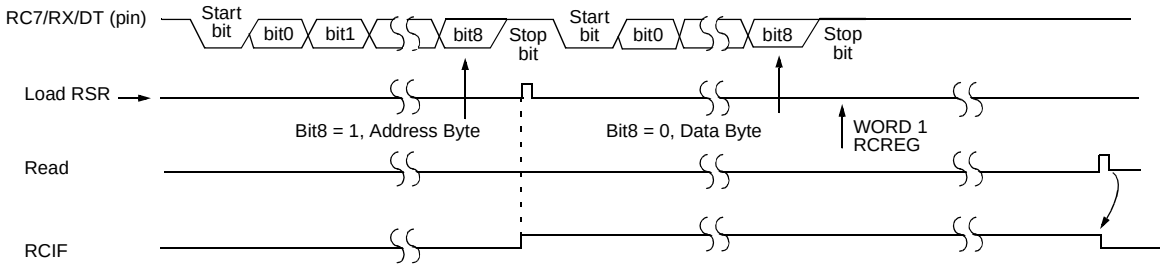


FIGURE 10-7: ASYNCHRONOUS RECEPTION WITH ADDRESS DETECT



Note: This timing diagram shows a data byte followed by an address byte. The data byte is not read into the RCREG (receive buffer) because ADDEN = 1.

FIGURE 10-8: ASYNCHRONOUS RECEPTION WITH ADDRESS BYTE FIRST



Note: This timing diagram shows a data byte followed by an address byte. The data byte is not read into the RCREG (receive buffer) because ADDEN was not updated and still = 0.

TABLE 10-7: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for Asynchronous Reception.
Note 1: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

© 1999 Microchip Technology Inc.

10.3 USART Synchronous Master Mode

In Synchronous Master mode, the data is transmitted in a half-duplex manne (i.e., transmission and reception do not occur at the same time). When transmitting data, the reception is inhibited and vice versa. Synchronous mode is entered by setting bit SYNC (TXSTA<4>). In addition, enable bit SPEN (RCSTA<7>) is set in order to configure the RC6/TX/CK and RC7/RX/DT I/O pins to CK (clock) and DT (data) lines respectively. The Master mode indicates that the processor transmits the master clock on the CK line. The Master mode is entered by setting bit CSRC (TXSTA<7>).

10.3.1 USART SYNCHRONOUS MASTER TRANSMISSION

The USART transmitter block diagram is shown in Figure 10-6. The heart of the transmitter is the transmit (serial) shift register (TSR). The shift register obtains its data from the read/write transmit buffer register TXREG. The TXREG register is loaded with data in software. The TSR register is not loaded until the last bit has been transmitted from the previous load. As soon as the last bit is transmitted, the TSR is loaded with new data from the TXREG (if available). Once the TXREG register transfers the data to the TSR register (occurs in one Tcycle), the TXREG is empty and interrupt bit TXIF (PIR1<4>) is set. The interrupt can be enabled/disabled by setting/clearing enable bit TXIE (PIE1<4>). Flag bit TXIF will be set regardless of the state of enable bit TXIE and cannot be cleared in software. It will reset only when new data is loaded into the TXREG register. While flag bit TXIF indicates the status of the TXREG register, another bit TRMT (TXSTA<1>) shows the status of the TSR register. TRMT is a read only bit which is set when the TSR is empty. No interrupt logic is tied to this bit, so the user has to poll this bit in order to determine if the TSR register is empty. The TSR is not mapped in data memory, so it is not available to the user.

Transmission is enabled by setting enable bit TXEN (TXSTA<5>). The actual transmission will not occur until the TXREG register has been loaded with data. The first data bit will be shifted out on the next available rising edge of the clock on the CK line. Data out is stable around the falling edge of the synchronous clock (Figure 10-9). The transmission can also be started by first loading the TXREG register and then setting bit TXEN (Figure 10-10). This is advantageous when slow baud rates are selected, since the BRG is kept in reset when bits TXEN, CREN and SREN are clear. Setting enable bit TXEN will start the BRG, creating a shift clock immediately. Normally, when transmission is first started, the TSR register is empty, so a transfer to the TXREG register will result in an immediate transfer to TSR resulting in an empty TXREG. Back-to-back transfers are possible.

Clearing enable bit TXEN during a transmission will cause the transmission to be aborted and will reset the transmitter. The DT and CK pins will revert to hi-impedance. If either bit CREN or bit SREN is set during a transmission, the transmission is aborted and the DT pin reverts to a hi-impedance state (for a reception). The CK pin will remain an output if bit CSRC is set (internal clock). The transmitter logic, however, is not reset, although it is disconnected from the pins. In order to reset the transmitter, the user has to clear bit TXEN. If bit SREN is set (to interrupt an on-going transmission and receive a single word), then after the single word is received, bit SREN will be cleared and the serial port will revert back to transmitting, since bit TXEN is still set. The DT line will immediately switch from hi-impedance receive mode to transmit and start driving. To avoid this, bit TXEN should be cleared.

In order to select 9-bit transmission, the TX9 (TXSTA<6>) bit should be set and the ninth bit should be written to bit TX9D (TXSTA<0>). The ninth bit must be written before writing the 8-bit data to the TXREG register. This is because a data write to the TXREG can result in an immediate transfer of the data to the TSR register (if the TSR is empty). If the TSR was empty and the TXREG was written before writing the “new” TX9D, the “present” value of bit TX9D is loaded.

Steps to follow when setting up a Synchronous Master Transmission:

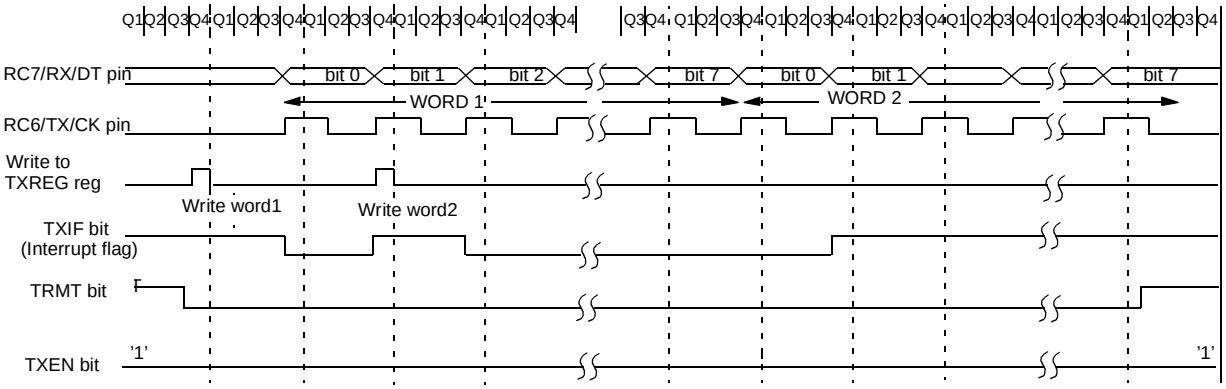
1. Initialize the SPBRG register for the appropriate baud rate (Section 10.1).
2. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
3. If interrupts are desired, set enable bit TXIE.
4. If 9-bit transmission is desired, set bit TX9.
5. Enable the transmission by setting bit TXEN.
6. If 9-bit transmission is selected, the ninth bit should be loaded in bit TX9D.
7. Start transmission by loading data to the TXREG register.

TABLE 10-8: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER TRANSMISSION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	USART Transmit Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

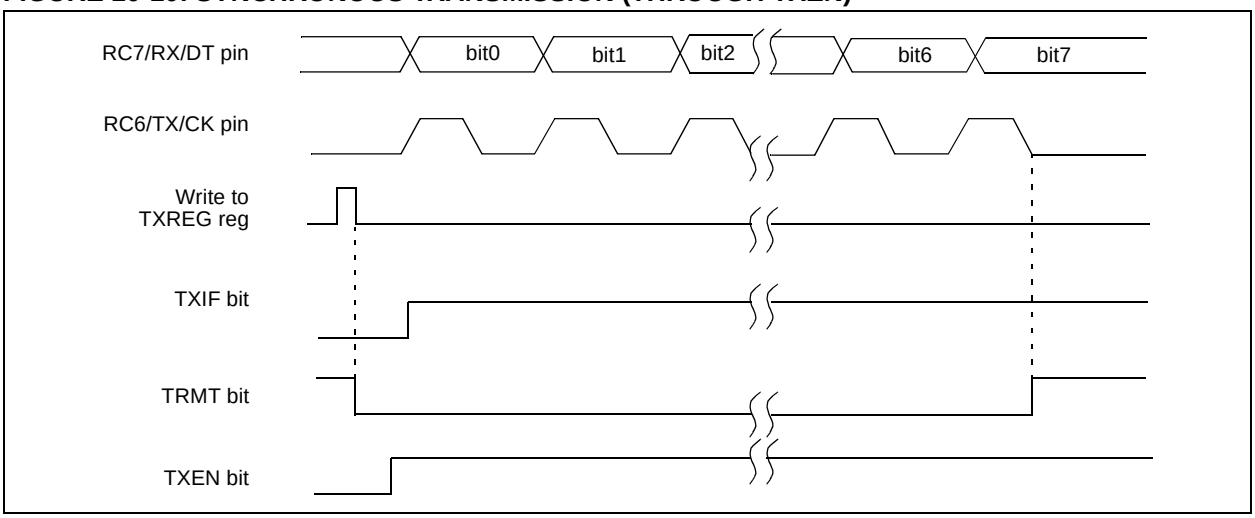
Legend: x = unknown, - = unimplemented, read as '0'. Shaded cells are not used for synchronous master transmission.
Note 1: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

FIGURE 10-9: SYNCHRONOUS TRANSMISSION



Note: Sync master mode; SPBRG = '0'. Continuous transmission of two 8-bit words

FIGURE 10-10: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)



© 1999 Microchip Technology Inc.

10.3.2 USART SYNCHRONOUS MASTER RECEPTION

Once synchronous mode is selected, reception is enabled by setting either enable bit SREN (RCSTA<5>) or enable bit CREN (RCSTA<4>). Data is sampled on the RC7/RX/DT pin on the falling edge of the clock. If enable bit SREN is set, then only a single word is received. If enable bit CREN is set, the reception is continuous until CREN is cleared. If both bits are set, CREN takes precedence. After clocking the last bit, the received data in the Receive Shift Register (RSR) is transferred to the RCREG register (if it is empty). When the transfer is complete, interrupt flag bit RCIF (PIR1<5>) is set. The actual interrupt can be enabled/disabled by setting/clearing enable bit RCIE (PIE1<5>). Flag bit RCIF is a read only bit, which is reset by the hardware. In this case, it is reset when the RCREG register has been read and is empty. The RCREG is a double buffered register (i.e., it is a two deep FIFO). It is possible for two bytes of data to be received and transferred to the RCREG FIFO and a third byte to begin shifting into the RSR register. On the clocking of the last bit of the third byte, if the RCREG register is still full, then overrun error bit OERR (RCSTA<1>) is set. The word in the RSR will be lost. The RCREG register can be read twice to retrieve the two bytes in the FIFO. Bit OERR has to be cleared in software (by clearing bit CREN). If bit OERR is set, transfers from the RSR to the RCREG are inhibited, so it is essential to clear bit OERR if it is set. The ninth receive bit is buffered the same way as the receive data. Reading the RCREG register will load bit RX9D with a new value, therefore it is essential for the user to read the RCSTA register before reading RCREG in order not to lose the old RX9D information.

Steps to follow when setting up a Synchronous Master Reception:

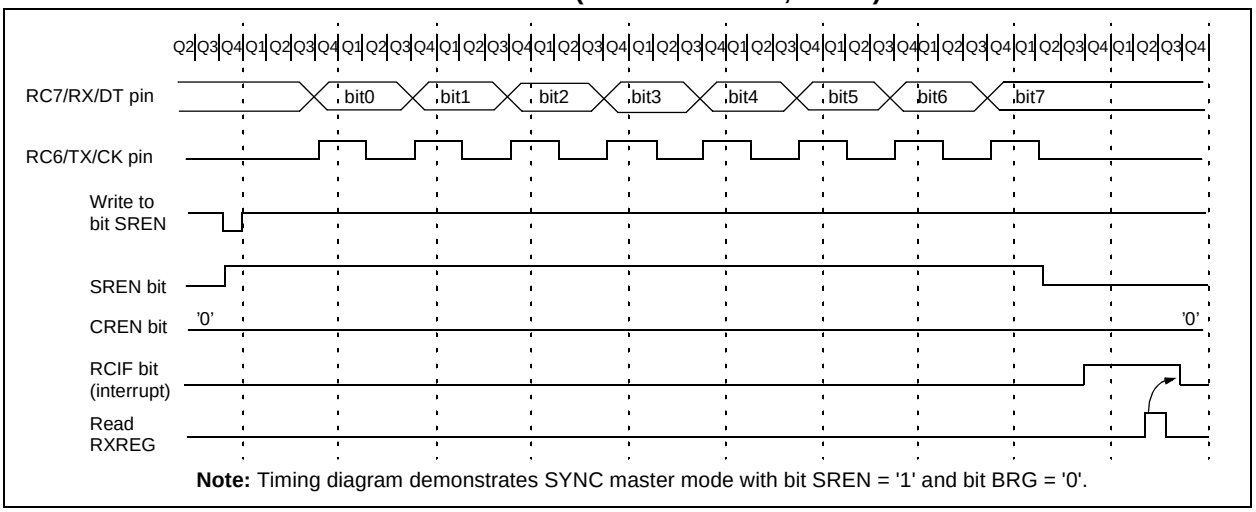
1. Initialize the SPBRG register for the appropriate baud rate. (Section 10.1)
2. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
3. Ensure bits CREN and SREN are clear.
4. If interrupts are desired, then set enable bit RCIE.
5. If 9-bit reception is desired, then set bit RX9.
6. If a single reception is required, set bit SREN. For continuous reception set bit CREN.
7. Interrupt flag bit RCIF will be set when reception is complete and an interrupt will be generated if enable bit RCIE was set.
8. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
9. Read the 8-bit received data by reading the RCREG register.
10. If any error occurred, clear the error by clearing bit CREN.

TABLE 10-9: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER RECEPTION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for synchronous master reception.
Note 1: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

FIGURE 10-11: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)



Note: Timing diagram demonstrates SYNC master mode with bit SREN = '1' and bit BRG = '0'.

10.4 USART Synchronous Slave Mode

Synchronous slave mode differs from the Master mode in the fact that the shift clock is supplied externally at the RC6/TX/CK pin (instead of being supplied internally in master mode). This allows the device to transfer or receive data while in SLEEP mode. Slave mode is entered by clearing bit CSRC (TXSTA<7>).

10.4.1 USART SYNCHRONOUS SLAVE TRANSMIT

The operation of the synchronous master and slave modes are identical except in the case of the SLEEP mode.

If two words are written to the TXREG and then the SLEEP instruction is executed, the following will occur:

a) The first word will immediately transfer to the TSR register and transmit.
b) The second word will remain in TXREG register.
c) Flag bit TXIF will not be set.
d) When the first word has been shifted out of TSR, the TXREG register will transfer the second word to the TSR and flag bit TXIF will now be set.
e) If enable bit TXIE is set, the interrupt will wake the chip from SLEEP and if the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

Steps to follow when setting up a Synchronous Slave Transmission:

1. Enable the synchronous slave serial port by setting bits SYNC and SPEN and clearing bit CSRC.
2. Clear bits CREN and SREN.
3. If interrupts are desired, then set enable bit TXIE.
4. If 9-bit transmission is desired, then set bit TX9.
5. Enable the transmission by setting enable bit TXEN.
6. If 9-bit transmission is selected, the ninth bit should be loaded in bit TX9D.
7. Start transmission by loading data to the TXREG register.

10.4.2 USART SYNCHRONOUS SLAVE RECEPTION

The operation of the synchronous master and slave modes is identical, except in the case of the SLEEP mode. Bit SREN is a “don't care” in slave mode.

If receive is enabled by setting bit CREN prior to the SLEEP instruction, then a word may be received during SLEEP. On completely receiving the word, the RSR register will transfer the data to the RCREG register and if enable bit RCIE bit is set, the interrupt generated will wake the chip from SLEEP. If the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

Steps to follow when setting up a Synchronous Slave Reception:

1. Enable the synchronous master serial port by setting bits SYNC and SPEN and clearing bit CSRC.
2. If interrupts are desired, set enable bit RCIE.
3. If 9-bit reception is desired, set bit RX9.
4. To enable reception, set enable bit CREN.
5. Flag bit RCIF will be set when reception is complete and an interrupt will be generated, if enable bit RCIE was set.
6. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
7. Read the 8-bit received data by reading the RCREG register.
8. If any error occurred, clear the error by clearing bit CREN.

© 1999 Microchip Technology Inc.

TABLE 10-10: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
19h	TXREG	USART Transmit Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Synchronous Slave Transmission.
Note 1: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

TABLE 10-11: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Synchronous Slave Reception.
Note 1: Bits PSPIE and PSPIF are reserved on the 28-pin devices, always maintain these bits clear.

NOTES:

© 1999 Microchip Technology Inc.

11.0 ANALOG-TO-DIGITAL CONVERTER (A/D) MODULE

The Analog-to-Digital (A/D) Converter module has five inputs for the 28-pin devices and eight for the other devices.

The analog input charges a sample and hold capacitor. The output of the sample and hold capacitor is the input into the converter. The converter then generates a digital result of this analog level via successive approximation. The A/D conversion of the analog input signal results in a corresponding 10-bit digital number. The A/D module has high and low voltage reference input that is software selectable to some combination of VDD, VSS, RA2 or RA3.

The A/D converter has a unique feature of being able to operate while the device is in SLEEP mode. To operate in sleep, the A/D clock must be derived from the A/D's internal RC oscillator.

The A/D module has four registers. These registers are:

- A/D Result High Register (ADRESH)
- A/D Result Low Register (ADRESL)
- A/D Control Register0 (ADCON0)
- A/D Control Register1 (ADCON1)

The ADCON0 register, shown in Register 11-1, controls the operation of the A/D module. The ADCON1 register, shown in Register 11-2, configures the functions of the port pins. The port pins can be configured as analog inputs (RA3 can also be the voltage reference) or as digital I/O.

Additional information on using the A/D module can be found in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).

REGISTER 11-1: ADCON0 REGISTER (ADDRESS: 1Fh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-6: **ADCS1:ADCS0: A/D Conversion Clock Select bits**
00 = FOSC/2
01 = FOSC/8
10 = FOSC/32
11 = FRC (clock derived from an RC oscillation)

bit 5-3: **CHS2:CHS0**: Analog Channel Select bits
000 = channel 0, (RA0/AN0)
001 = channel 1, (RA1/AN1)
010 = channel 2, (RA2/AN2)
011 = channel 3, (RA3/AN3)
100 = channel 4, (RA5/AN4)
101 = channel 5, (RE0/AN5)(1)
110 = channel 6, (RE1/AN6)(1)
111 = channel 7, (RE2/AN7)(1)

bit 2: **GO/$\overline{\text{DONE}}$:** A/D Conversion Status bit
If ADON = 1
1 = A/D conversion in progress (setting this bit starts the A/D conversion)
0 = A/D conversion not in progress (This bit is automatically cleared by hardware when the A/D conversion is complete)

bit 1: **Unimplemented**: Read as '0'

bit 0: **ADON**: A/D On bit
1 = A/D converter module is operating
0 = A/D converter module is shutoff and consumes no operating current

Note 1: These channels are not available on the 28-pin devices.

REGISTER 11-2: ADCON1 REGISTER (ADDRESS 9Fh)

U-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **ADFM: A/D Result format select**
1 = Right Justified. 6 most significant bits of ADRESH are read as '0'.
0 = Left Justified. 6 least significant bits of ADRESL are read as '0'.

bit 6-4: **Unimplemented: Read as '0'**

bit 3-0: **PCFG3:PCFG0**: A/D Port Configuration Control bits

PCFG3: PCFG0	AN7(1) RE2	AN6(1) RE1	AN5(1) RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	CHAN / Refs(2)
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	RA3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	RA3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	RA3	VSS	2/1
011x	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	RA3	RA2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	RA3	RA2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	RA3	RA2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	RA3	RA2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	RA3	RA2	1/2

A = Analog input

D = Digital I/O

Note 1: These channels are not available on the 28-pin devices.

2: This column indicates the number of analog channels available as A/D inputs and the numer of analog channels used as voltage reference inputs.

© 1999 Microchip Technology Inc.

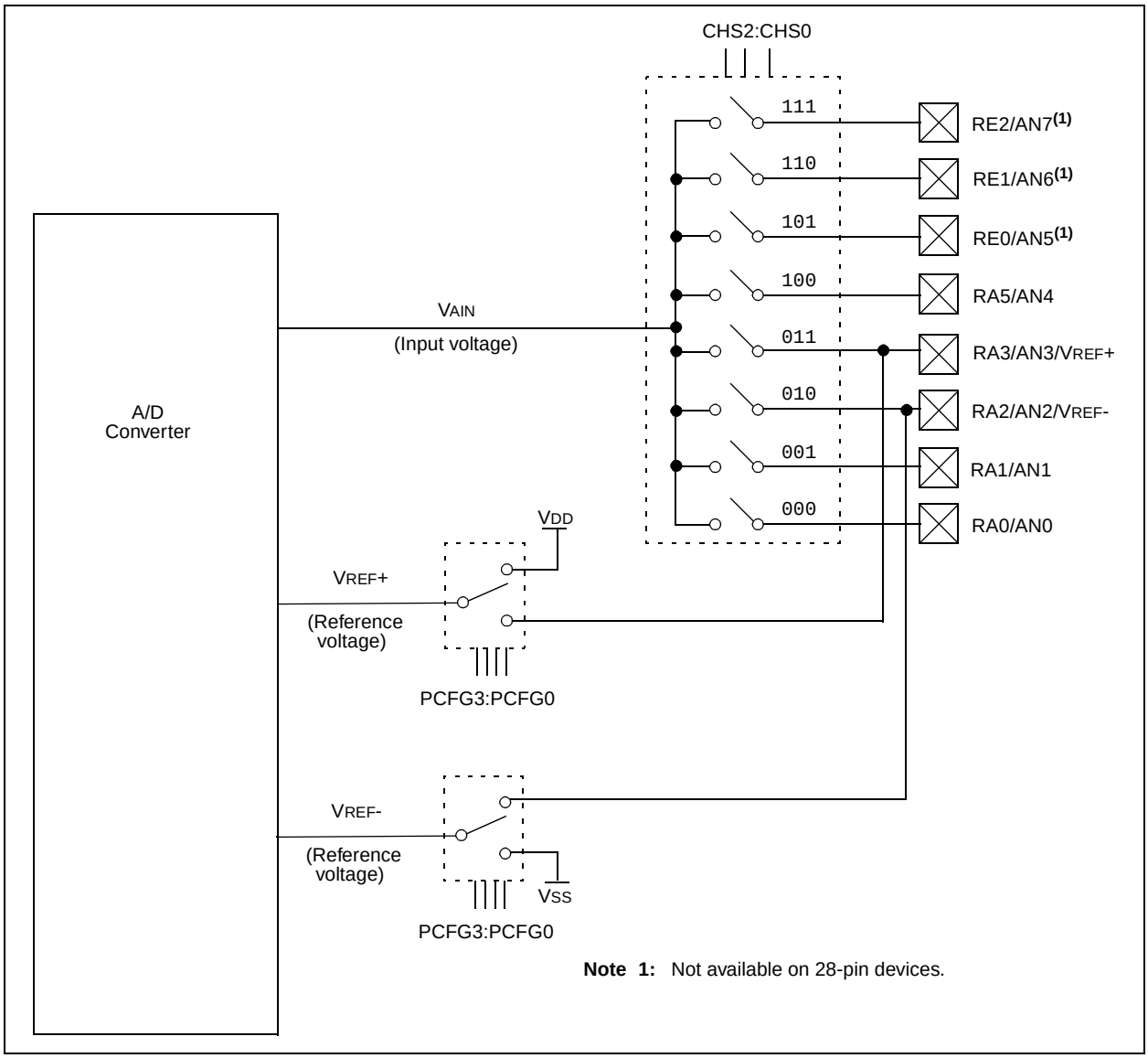
The ADRESH:ADRESL registers contain the 10-bit result of the A/D conversion. When the A/D conversion is complete, the result is loaded into this A/D result register pair, the GO/$\overline{\text{DONE}}$ bit (ADCON0<2>) is cleared and the A/D interrupt flag bit ADIF is set. The block diagram of the A/D module is shown in Figure 11-1.

After the A/D module has been configured as desired, the selected channel must be acquired before the conversion is started. The analog input channels must have their corresponding TRIS bits selected as inputs. To determine sample time, see Section 11.1. After this acquisition time has elapsed, the A/D conversion can be started. The following steps should be followed for doing an A/D conversion:

1. Configure the A/D module:
   - Configure analog pins / voltage reference / and digital I/O (ADCON1)
   - Select A/D input channel (ADCON0)
   - Select A/D conversion clock (ADCON0)
   - Turn on A/D module (ADCON0)
2. Configure A/D interrupt (if desired):
   - Clear ADIF bit
   - Set ADIE bit
   - Set GIE bit
3. Wait the required acquisition time.
4. Start conversion:
   - Set GO/$\overline{\text{DONE}}$ bit (ADCON0)
5. Wait for A/D conversion to complete, by either:
   - Polling for the GO/$\overline{\text{DONE}}$ bit to be cleared

   OR
   - Waiting for the A/D interrupt
6. Read A/D Result register pair (ADRESH:ADRESL), clear bit ADIF if required.
7. For next conversion, go to step 1 or step 2 as required. The A/D conversion time per bit is defined as T_{AD}. A minimum wait of $2T_{AD}$ is required before next acquisition starts.

FIGURE 11-1: A/D BLOCK DIAGRAM



Note 1: Not available on 28-pin devices.

11.1 A/D Acquisition Requirements

For the A/D converter to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The analog input model is shown in Figure 11-2. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), Figure 11-2. **The maximum recommended impedance for analog sources is 10 kΩ.** As the impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (changed), this acquisition must be done before the conversion can be started.

To calculate the minimum acquisition time, Equation 11-1 may be used. This equation assumes that 1/2 LSb error is used (1024 steps for the A/D). The 1/2 LSb error is the maximum error allowed for the A/D to meet its specified resolution.

To calculate the minimum acquisition time, TACQ, see the PICmicro™ Mid-Range Reference Manual (DS33023).

© 1999 Microchip Technology Inc.

EQUATION 11-1: ACQUISITION TIME

T_{ACQ}	=	Amplifier Settling Time +
		Hold Capacitor Charging Time +
		Temperature Coefficient
	=	$T_{AMP} + T_C + T_{COFF}$
	=	$2\mu S + T_C + [(\text{Temperature } -25°C)(0.05\mu S/°C)]$
T_C	=	$C_{HOLD}(R_{IC} + R_{SS} + R_S) \ln(1/2047)$
	=	$-120pF (1k\Omega + 7k\Omega + 10k\Omega) \ln(0.0004885)$
	=	$16.47\mu S$
T_{ACQ}	=	$2\mu S + 16.47\mu S + [(50°C -25\times C)(0.05\mu S/\times C)$
	=	$19.72\mu S$

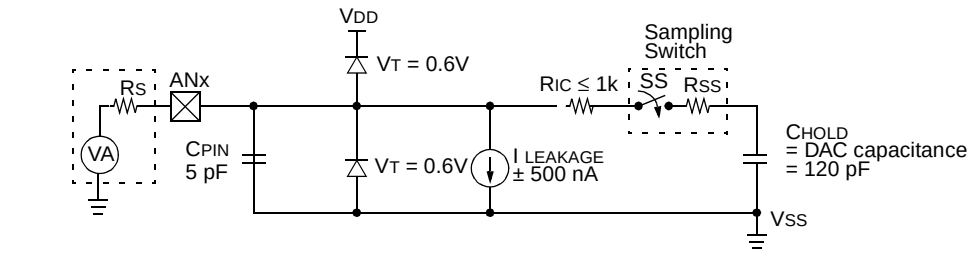
Note 1: The reference voltage (V_{REF}) has no effect on the equation, since it cancels itself out.

2: The charge holding capacitor (C_{HOLD}) is not discharged after each conversion.

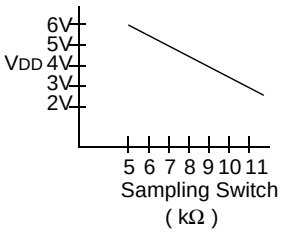
3: The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

4: After a conversion has completed, a 2.0T_{AD} delay must complete before acquisition can begin again. During this time, the holding capacitor is not connected to the selected A/D input channel.

FIGURE 11-2: ANALOG INPUT MODEL



Legend		
	C_{PIN}	= input capacitance
	V_T	= threshold voltage
	$I_{LEAKAGE}$	= leakage current at the pin due to various junctions
	R_{IC}	= interconnect resistance
	SS	= sampling switch
	C_{HOLD}	= sample/hold capacitance (from DAC)



11.2 Selecting the A/D Conversion Clock

The A/D conversion time per bit is defined as TAD. The A/D conversion requires a minimum 12TAD per 10-bit conversion. The source of the A/D conversion clock is software selected. The four possible options for TAD are:

- 2TOSC
- 8TOSC
- 32TOSC
- Internal RC oscillator

For correct A/D conversions, the A/D conversion clock (TAD) must be selected to ensure a minimum TAD time of 1.6 µs.

Table 11-1shows the resultant TAD times derived from the device operating frequencies and the A/D clock source selected.

TABLE 11-1: TAD vs. MAXIMUM DEVICE OPERATING FREQUENCIES (STANDARD DEVICES (C))

AD Clock Source (TAD)		Maximum Device Frequency
Operation	**ADCS1:ADCS0**	**Max.**
2TOSC	00	1.25 MHz
8TOSC	01	5 MHz
32TOSC	10	20 MHz
RC(1, 2, 3)	11	Note 1

Note 1: The RC source has a typical TAD time of 4 µs but can vary between 2-6 µs.
2: When the device frequencies are greater than 1 MHz, the RC A/D conversion clock source is only recommended for sleep operation.
3: For extended voltage devices (LC), please refer to the Electrical Specifications section.

11.3 Configuring Analog Port Pins

The ADCON1, and TRIS registers control the operation of the A/D port pins. The port pins that are desired as analog inputs must have their corresponding TRIS bits set (input). If the TRIS bit is cleared (output), the digital output level (VOH or VOL) will be converted.

The A/D operation is independent of the state of the CHS2:CHS0 bits and the TRIS bits.

Note 1: When reading the port register, any pin configured as an analog input channel will read as cleared (a low level). Pins configured as digital inputs will convert an analog input. Analog levels on a digitally configured input will not affect the conversion accuracy.

2: Analog levels on any pin that is defined as a digital input (including the AN7:AN0 pins), may cause the input buffer to consume current that is out of the device specifications.

© 1999 Microchip Technology Inc.

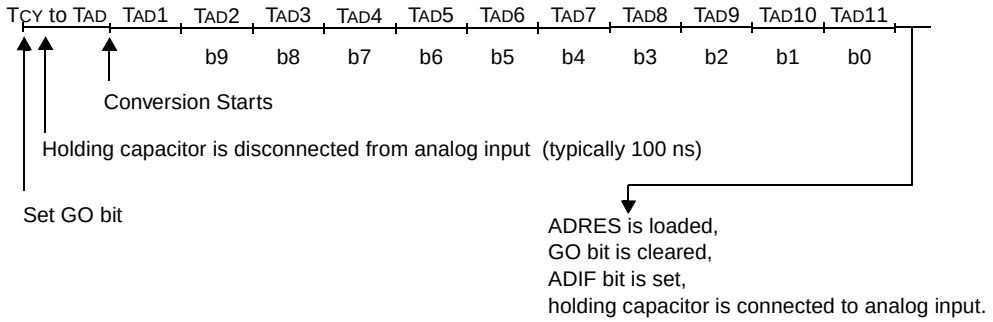
11.4 A/D Conversions

Clearing the GO/$\overline{\text{DONE}}$ bit during a conversion will abort the current conversion. The A/D result register pair will NOT be updated with the partially completed A/D conversion sample. That is, the ADRESH:ADRESL registers will continue to contain the value of the last completed conversion (or the last value written to the ADRESH:ADRESL registers). After the A/D conversion is aborted, a 2TAD wait is required before the next acquisition is started. After this 2TAD wait, acquisition on the selected channel is automatically started.

In Figure 11-3, after the GO bit is set, the first time segmant has a minimum of TCY and a maximum of TAD.

Note:	The GO/$\overline{\text{DONE}}$ bit should **NOT** be set in the same instruction that turns on the A/D.

FIGURE 11-3: A/D CONVERSION TAD CYCLES



11.4.1 A/D RESULT REGISTERS

The ADRESH:ADRESL register pair is the location where the 10-bit A/D result is loaded at the completion of the A/D conversion. This register pair is 16-bits wide. The A/D module gives the flexibility to left or right justify the 10-bit result in the 16-bit result register. The A/D Format Select bit (ADFM) controls this justification. Figure 11-4 shows the operation of the A/D result justification. The extra bits are loaded with '0's'. When an A/D result will not overwrite these locations (A/D disable), these registers may be used as two general purpose 8-bit registers.

11.5 A/D Operation During Sleep

The A/D module can operate during SLEEP mode. This requires that the A/D clock source be set to RC (ADCS1:ADCS0 = 11). When the RC clock source is selected, the A/D module waits one instruction cycle before starting the conversion. This allows the SLEEP instruction to be executed, which eliminates all digital switching noise from the conversion. When the conversion is completed the GO/$\overline{\text{DONE}}$ bit will be cleared and the result loaded into the ADRES register. If the A/D interrupt is enabled, the device will wake-up from SLEEP. If the A/D interrupt is not enabled, the A/D module will then be turned off, although the ADON bit will remain set.

When the A/D clock source is another clock option (not RC), a SLEEP instruction will cause the present conversion to be aborted and the A/D module to be turned off, though the ADON bit will remain set.

Turning off the A/D places the A/D module in its lowest current consumption state.

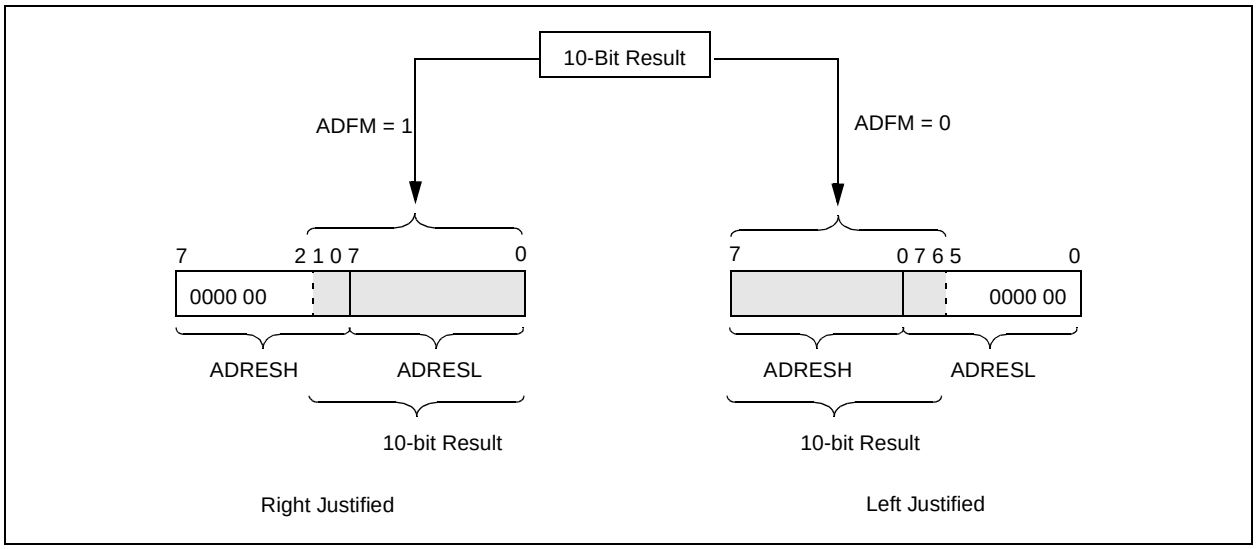
Note:	For the A/D module to operate in SLEEP, the A/D clock source must be set to RC (ADCS1:ADCS0 = 11). To allow the conversion to occur during SLEEP, ensure the SLEEP instruction immediately follows the instruction that sets the GO/$\overline{\text{DONE}}$ bit.

11.6 Effects of a Reset

A device reset forces all registers to their reset state. This forces the A/D module to be turned off, and any conversion is aborted.

The value that is in the ADRESH:ADRESL registers is not modified for a Power-on Reset. The ADRESH:ADRESL registers will contain unknown data after a Power-on Reset.

FIGURE 11-4: A/D RESULT JUSTIFICATION



© 1999 Microchip Technology Inc.

TABLE 11-2: REGISTERS/BITS ASSOCIATED WITH A/D

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR, BOR	$\overline{MCLR}$, WDT
0Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
1Eh	ADRESH	A/D Result Register High Byte								xxxx xxxx	uuuu uuuu
9Eh	ADRESL	A/D Result Register Low Byte								xxxx xxxx	uuuu uuuu
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	—	ADON	0000 00-0	0000 00-0
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--0x 0000	--0u 0000
89h(1)	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111
09h(1)	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used for A/D conversion.
Note 1: These registers/bits are not available on the 28-pin devices.

NOTES:

© 1999 Microchip Technology Inc.

12.0 SPECIAL FEATURES OF THE CPU

These devices have a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These are:

- OSC Selection
- Reset
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- SLEEP
- Code protection
- ID locations
- In-Circuit Serial Programming
- Low Voltage In-Circuit Serial Programming
- In-Circuit Debugger

These devices have a watchdog timer, which can be shut off only through configuration bits. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in reset until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay of 72 ms (nominal) on power-up only. It is designed to keep the part in reset while the power supply stabilizes. With these two timers on-chip, most applications need no external reset circuitry.

SLEEP mode is designed to offer a very low current power-down mode. The user can wake-up from SLEEP through external reset, Watchdog Timer Wake-up, or through an interrupt. Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost while the LP crystal option saves power. A set of configuration bits are used to select various options.

Additional information on special features is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

12.1 Configuration Bits

The configuration bits can be programmed (read as '0') or left unprogrammed (read as '1') to select various device configurations. These bits are mapped in program memory location 2007h.

The user will note that address 2007h is beyond the user program memory space. In fact, it belongs to the special test/configuration memory space (2000h - 3FFFh), which can be accessed only during programming.

REGISTER 12-1: CONFIGURATION WORD

CP1	CP0	DEBUG	—	WRT	CPD	LVP	BODEN	CP1	CP0	$\overline{\text{PWRTE}}$	WDTE	F0SC1	F0SC0
bit13													bit0

Register: CONFIG
Address 2007h

bit 13-12:

bit 5-4: **CP1:CP0:** Flash Program Memory Code Protection bits **(2)**
- 11 = Code protection off
- 10 = 1F00h to 1FFFh code protected (PIC16F877, 876)
- 10 = 0F00h to 0FFFh code protected (PIC16F874, 873)
- 01 = 1000h to 1FFFh code protected (PIC16F877, 876)
- 01 = 0800h to 0FFFh code protected (PIC16F874, 873)
- 00 = 0000h to 1FFFh code protected (PIC16F877, 876)
- 00 = 0000h to 0FFFh code protected (PIC16F874, 873)

bit 11: **DEBUG:** In-Circuit Debugger Mode
- 1 = In-Circuit Debugger disabled, RB6 and RB7 are general purpose I/O pins.
- 0 = In-Circuit Debugger enabled, RB6 and RB7 are dedicated to the debugger.

bit 10: **Unimplemented: Read as '1'**

bit 9: **WRT:** Flash Program Memory Write Enable
- 1 = Unprotected program memory may be written to by EECON control
- 0 = Unprotected program memory may not be written to by EECON control

bit 8: **CPD:** Data EE Memory Code Protection
- 1 = Code protection off
- 0 = Data EEPROM memory code protected

bit 7: **LVP**: Low Voltage In-Circuit Serial Programming Enable bit
- 1 = RB3/PGM pin has PGM function, low voltage programming enabled
- 0 = RB3 is digital I/O, HV on $\overline{\text{MCLR}}$ must be used for programming

bit 6: **BODEN**: Brown-out Reset Enable bit **(1)**
- 1 = BOR enabled
- 0 = BOR disabled

bit 3: **$\overline{\text{PWRTE}}$**: Power-up Timer Enable bit **(1)**
- 1 = PWRT disabled
- 0 = PWRT enabled

bit 2: **WDTE**: Watchdog Timer Enable bit
- 1 = WDT enabled
- 0 = WDT disabled

bit 1-0: **FOSC1:FOSC0**: Oscillator Selection bits
- 11 = RC oscillator
- 10 = HS oscillator
- 01 = XT oscillator
- 00 = LP oscillator

Note 1: Enabling Brown-out Reset automatically enables Power-up Timer (PWRT), regardless of the value of bit $\overline{\text{PWRTE}}$. Ensure the Power-up Timer is enabled anytime Brown-out Reset is enabled.

2: All of the CP1:CP0 pairs have to be given the same value to enable the code protection scheme listed.

© 1999 Microchip Technology Inc.

12.2 Oscillator Configurations

12.2.1 OSCILLATOR TYPES

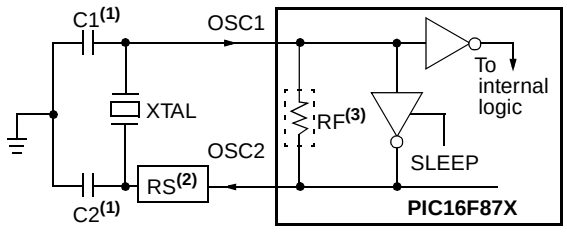
The PIC16F87X can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1 and FOSC0) to select one of these four modes:

- LP Low Power Crystal
- XT Crystal/Resonator
- HS High Speed Crystal/Resonator
- RC Resistor/Capacitor

12.2.2 CRYSTAL OSCILLATOR/CERAMIC RESONATORS

In XT, LP or HS modes, a crystal or ceramic resonator is connected to the OSC1/CLKIN and OSC2/CLKOUT pins to establish oscillation (Figure 12-1). The PIC16F87X oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications. When in XT, LP or HS modes, the device can have an external clock source to drive the OSC1/CLKIN pin (Figure 12-2).

FIGURE 12-1: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)



Note 1: See Table 12-1 and Table 12-2 for recommended values of C1 and C2.

2: A series resistor (RS) may be required for AT strip cut crystals.

3: RF varies with the crystal chosen.

FIGURE 12-2: EXTERNAL CLOCK INPUT OPERATION (HS, XT OR LP OSC CONFIGURATION)

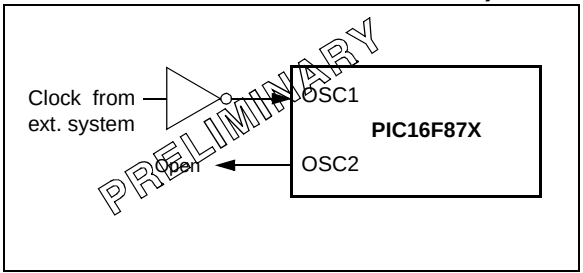


TABLE 12-1: CERAMIC RESONATORS

Ranges Tested:			
Mode	**Freq**	**OSC1**	**OSC2**
XT	455 kHz 2.0 MHz 4.0 MHz	68 - 100 pF 15 - 68 pF 15 - 68 pF	68 - 100 pF 15 - 68 pF 15 - 68 pF
HS	8.0 MHz 16.0 MHz	10 - 68 pF 10 - 22 pF	10 - 68 pF 10 - 22 pF
These values are for design guidance only. See notes at bottom of page.			
Resonators Used:			
455 kHz	Panasonic EFO-A455K04B	± 0.3%	
2.0 MHz	Murata Erie CSA2.00MG	± 0.5%	
4.0 MHz	Murata Erie CSA4.00MG	± 0.5%	
8.0 MHz	Murata Erie CSA8.00MT	± 0.5%	
16.0 MHz	Murata Erie CSA16.00MX	± 0.5%	
All resonators used did not have built-in capacitors.			

PRELIMINARY

TABLE 12-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR

Osc Type	Crystal Freq	Cap. Range C1	Cap. Range C2
LP	32 kHz	33 pF	33 pF
	200 kHz	15 pF	15 pF
XT	200 kHz	47-68 pF	47-68 pF
	1 MHz	15 pF	15 pF
	4 MHz	15 pF	15 pF
HS	4 MHz	15 pF	15 pF
	8 MHz	15-33 pF	15-33 pF
	20 MHz	15-33 pF	15-33 pF
These values are for design guidance only. See notes at bottom of page.			
Crystals Used			
32 kHz	Epson C-001R32.768K-A	± 20 PPM	
200 kHz	STD XTL 200.000KHz	± 20 PPM	
1 MHz	ECS ECS-10-13-1	± 50 PPM	
4 MHz	ECS ECS-40-20-1	± 50 PPM	
8 MHz	EPSON CA-301 8.000M-C	± 30 PPM	
20 MHz	EPSON CA-301 20.000M-C	± 30 PPM	

Note 1: Higher capacitance increases the stability of oscillator but also increases the start-up time.

2: Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.

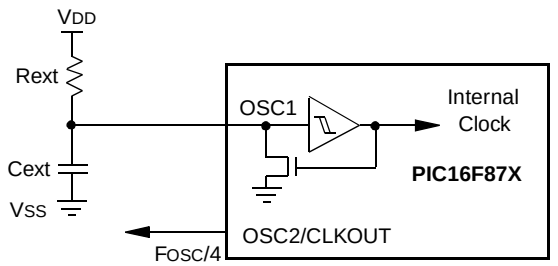
3: Rs may be required in HS mode, as well as XT mode, to avoid overdriving crystals with low drive level specification.

4: When migrating from other PICmicro devices, oscillator performance should be verified.

12.2.3 RC OSCILLATOR

For timing insensitive applications, the “RC” device option offers additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor (REXT) and capacitor (CEXT) values, and the operating temperature. In addition to this, the oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low CEXT values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 12-3 shows how the R/C combination is connected to the PIC16F87X.

FIGURE 12-3: RC OSCILLATOR MODE



Recommended values: 3 kΩ ≤ Rext ≤ 100 kΩ
Cext > 20pF

© 1999 Microchip Technology Inc.

12.3 Reset

The PIC16F87X differentiates between various kinds of reset:

- Power-on Reset (POR)
- $\overline{MCLR}$ reset during normal operation
- $\overline{MCLR}$ reset during SLEEP
- WDT Reset (during normal operation)
- WDT Wake-up (during SLEEP)
- Brown-out Reset (BOR)

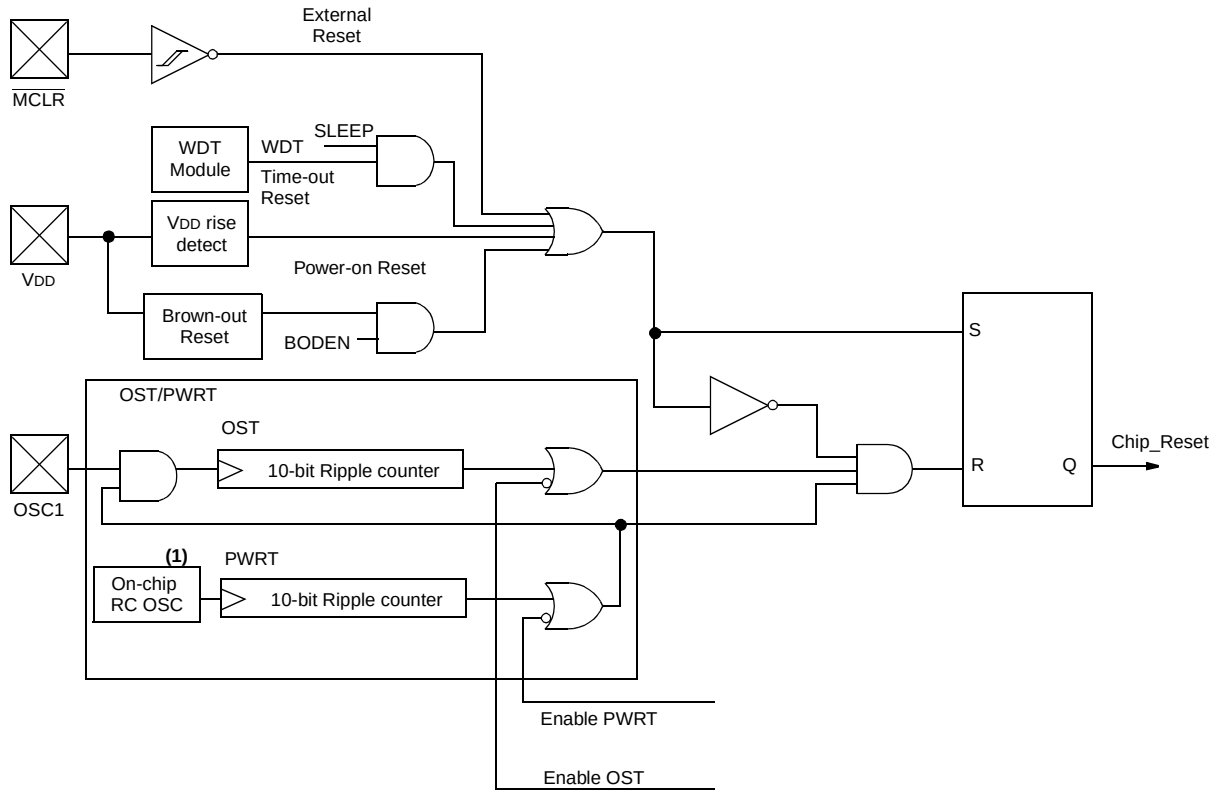
Some registers are not affected in any reset condition. Their status is unknown on POR and unchanged in any other reset. Most other registers are reset to a "reset state" on Power-on Reset (POR), on the $\overline{MCLR}$ and WDT Reset, on $\overline{MCLR}$ reset during SLEEP, and Brown-out Reset (BOR). They are not affected by a WDT Wake-up, which is viewed as the resumption of normal operation. The $\overline{TO}$ and $\overline{PD}$ bits are set or cleared differently in different reset situations as indicated in Table 12-4. These bits are used in software to determine the nature of the reset. See Table 12-6 for a full description of reset states of all registers.

A simplified block diagram of the on-chip reset circuit is shown in Figure 12-4.

These devices have a $\overline{MCLR}$ noise filter in the $\overline{MCLR}$ reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive $\overline{MCLR}$ pin low.

FIGURE 12-4: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



Note 1: This is a separate oscillator from the RC oscillator of the CLKIN pin.

12.4 Power-On Reset (POR)

A Power-on Reset pulse is generated on-chip when VDD rise is detected (in the range of 1.2V - 1.7V). To take advantage of the POR, tie the $\overline{MCLR}$ pin directly (or through a resistor) to VDD. This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for VDD is specified. See Electrical Specifications for details.

When the device starts normal operation (exits the reset condition), device operating parameters (voltage, frequency, temperature,...) must be met to ensure operation. If these conditions are not met, the device must be held in reset until the operating conditions are met. Brown-out Reset may be used to meet the start-up conditions. For additional information, refer to Application Note, AN007, “Power-up Trouble Shooting”, (DS00007).

12.5 Power-up Timer (PWRT)

The Power-up Timer provides a fixed 72 ms nominal time-out on power-up only from the POR. The Power-up Timer operates on an internal RC oscillator. The chip is kept in reset as long as the PWRT is active. The PWRT’s time delay allows VDD to rise to an acceptable level. A configuration bit is provided to enable/disable the PWRT.

The power-up time delay will vary from chip to chip due to VDD, temperature and process variation. See DC parameters for details (TPWRT, parameter #33).

12.6 Oscillator Start-up Timer (OST)

The Oscillator Start-up Timer (OST) provides 1024 oscillator cycle (from OSC1 input) delay after the PWRT delay is over. This ensures that the crystal oscillator or resonator has started and stabilized.

The OST time-out is invoked only for XT, LP and HS modes and only on Power-on Reset or wake-up from SLEEP.

12.7 Brown-Out Reset (BOR)

The configuration bit, BODEN, can enable or disable the Brown-out Reset circuit. If VDD falls below VBOR (parameter D005, about 4V) for longer than TBOR (parameter #35, about 100μS), the brown-out situation will reset the device. If VDD falls below VBOR for less than TBOR, a reset may not occur.

Once the brown-out occurs, the device will remain in brown-out reset until VDD rises above VBOR. The power-up timer then keeps the device in reset for TPWRT (parameter #33, about 72mS). If VDD should fall below VBOR during TPWRT, the brown-out reset process will restart when VDD rises above VBOR with the power-up timer reset. The power-up timer is always enabled when the brown-out reset circuit is enabled regardless of the state of the PWRT configuration bit.

12.8 Time-out Sequence

On power-up, the time-out sequence is as follows: The PWRT delay starts (if enabled) when a POR reset occurs. Then OST starts counting 1024 oscillator cycles when PWRT ends (LP, XT, HS). When the OST ends, the device comes out of RESET.

If $\overline{MCLR}$ is kept low long enough, the time-outs will expire. Bringing $\overline{MCLR}$ high will begin execution immediately. This is useful for testing purposes or to synchronize more than one PIC16CXX device operating in parallel.

Table 12-5 shows the reset conditions for the STATUS, PCON and PC registers, while Table 12-6 shows the reset conditions for all the registers.

12.9 Power Control/Status Register (PCON)

The Power Control/Status Register, PCON, has up to two bits depending upon the device.

Bit0 is Brown-out Reset Status bit, $\overline{BOR}$. Bit $\overline{BOR}$ is unknown on a Power-on Reset. It must then be set by the user and checked on subsequent resets to see if bit $\overline{BOR}$ cleared, indicating a BOR occurred. The $\overline{BOR}$ bit is a "don’t care" bit and is not necessarily predictable if the Brown-out Reset circuitry is disabled (by clearing bit BODEN in the Configuration Word).

Bit1 is $\overline{POR}$ (Power-on Reset Status bit). It is cleared on a Power-on Reset and unaffected otherwise. The user must set this bit following a Power-on Reset.

© 1999 Microchip Technology Inc.

TABLE 12-3: TIME-OUT IN VARIOUS SITUATIONS

Oscillator Configuration	Power-up		Brown-out	Wake-up from SLEEP
	$\overline{PWRTE}$ = 0	$\overline{PWRTE}$ = 1		
XT, HS, LP	72 ms + 1024TOSC	1024TOSC	72 ms + 1024TOSC	1024TOSC
RC	72 ms	—	72 ms	—

TABLE 12-4: STATUS BITS AND THEIR SIGNIFICANCE

POR	BOR	TO	PD	
0	x	1	1	Power-on Reset
0	x	0	x	Illegal, $\overline{TO}$ is set on $\overline{POR}$
0	x	x	0	Illegal, $\overline{PD}$ is set on $\overline{POR}$
1	0	1	1	Brown-out Reset
1	1	0	1	WDT Reset
1	1	0	0	WDT Wake-up
1	1	u	u	$\overline{MCLR}$ Reset during normal operation
1	1	1	0	$\overline{MCLR}$ Reset during SLEEP or interrupt wake-up from SLEEP

TABLE 12-5: RESET CONDITION FOR SPECIAL REGISTERS

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	000h	0001 1xxx	---- --0x
$\overline{MCLR}$ Reset during normal operation	000h	000u uuuu	---- --uu
$\overline{MCLR}$ Reset during SLEEP	000h	0001 0uuu	---- --uu
WDT Reset	000h	0000 1uuu	---- --uu
WDT Wake-up	PC + 1	uuu0 0uuu	---- --uu
Brown-out Reset	000h	0001 1uuu	---- --u0
Interrupt wake-up from SLEEP	PC + 1(1)	uuu1 0uuu	---- --uu

Legend: u = unchanged, x = unknown, - = unimplemented bit read as '0'.

Note 1: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

TABLE 12-6: INITIALIZATION CONDITIONS FOR ALL REGISTERS

Register	Devices				Power-on Reset, Brown-out Reset	$\overline{MCLR}$ Resets WDT Reset	Wake-up via WDT or Interrupt
W	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	873	874	876	877	N/A	N/A	N/A
TMR0	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	873	874	876	877	0000h	0000h	PC + 1(2)
STATUS	873	874	876	877	0001 1xxx	000q quuu(3)	uuuq quuu(3)
FSR	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA	873	874	876	877	--0x 0000	--0u 0000	--uu uuuu
PORTB	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTC	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTD	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTE	873	874	876	877	---- -xxx	---- -uuu	---- -uuu
PCLATH	873	874	876	877	---0 0000	---0 0000	---u uuuu
INTCON	873	874	876	877	0000 000x	0000 000u	uuuu uuuu(1)
PIR1	873	874	876	877	r000 0000	r000 0000	ruuu uuuu(1)
	873	874	876	877	0000 0000	0000 0000	uuuu uuuu(1)
PIR2	873	874	876	877	-r-0 0--0	-r-0 0--0	-r-u u--u(1)
TMR1L	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	873	874	876	877	--00 0000	--uu uuuu	--uu uuuu
TMR2	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
T2CON	873	874	876	877	-000 0000	-000 0000	-uuu uuuu
SSPBUF	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
SSPCON	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
CCPR1L	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	873	874	876	877	--00 0000	--00 0000	--uu uuuu
RCSTA	873	874	876	877	0000 000x	0000 000x	uuuu uuuu
TXREG	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
RCREG	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
CCPR2L	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR2H	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP2CON	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
ADRESH	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	873	874	876	877	0000 00-0	0000 00-0	uuuu uu-u
OPTION_REG	873	874	876	877	1111 1111	1111 1111	uuuu uuuu
TRISA	873	874	876	877	--11 1111	--11 1111	--uu uuuu
TRISB	873	874	876	877	1111 1111	1111 1111	uuuu uuuu
TRISC	873	874	876	877	1111 1111	1111 1111	uuuu uuuu
TRISD	873	874	876	877	1111 1111	1111 1111	uuuu uuuu
TRISE	873	874	876	877	0000 -111	0000 -111	uuuu -uuu
PIE1	873	874	876	877	r000 0000	r000 0000	ruuu uuuu
	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
PIE2	873	874	876	877	-r-0 0--0	-r-0 0--0	-r-u u--u

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition, r = reserved maintain clear.

Note 1: One or more bits in INTCON, PIR1 and/or PIR2 will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

3: See Table 12-5 for reset value for specific condition.

© 1999 Microchip Technology Inc.

TABLE 12-6: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)

Register	Devices				Power-on Reset, Brown-out Reset	MCLR Resets WDT Reset	Wake-up via WDT or Interrupt
PCON	873	874	876	877	---- --qq	---- --uu	---- --uu
PR2	873	874	876	877	1111 1111	1111 1111	1111 1111
SSPADD	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
SSPSTAT	873	874	876	877	--00 0000	--00 0000	--uu uuuu
TXSTA	873	874	876	877	0000 -010	0000 -010	uuuu -uuu
SPBRG	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
ADRESL	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON1	873	874	876	877	0--- 0000	0--- 0000	u--- uuuu
EEDATA	873	874	876	877	0--- 0000	0--- 0000	u--- uuuu
EEADR	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
EEDATH	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
EEADRH	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
EECON1	873	874	876	877	x--- x000	u--- u000	u--- uuuu
EECON2	873	874	876	877	---- ----	---- ----	---- ----

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition, r = reserved maintain clear.

Note 1: One or more bits in INTCON, PIR1 and/or PIR2 will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

3: See Table 12-5 for reset value for specific condition.

FIGURE 12-5: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ TIED TO V_{DD})

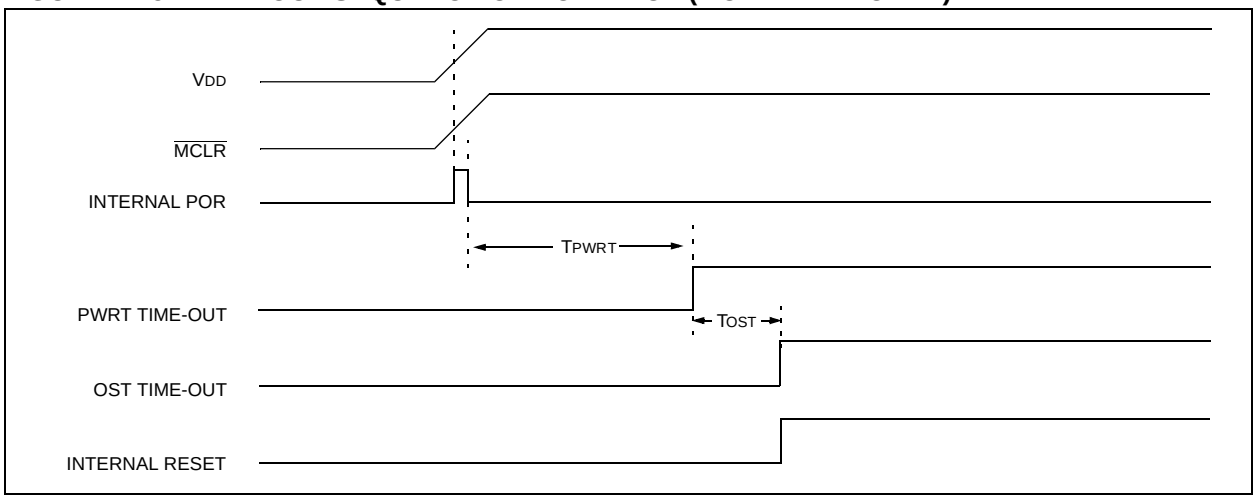


FIGURE 12-6: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ NOT TIED TO V_{DD}): CASE 1

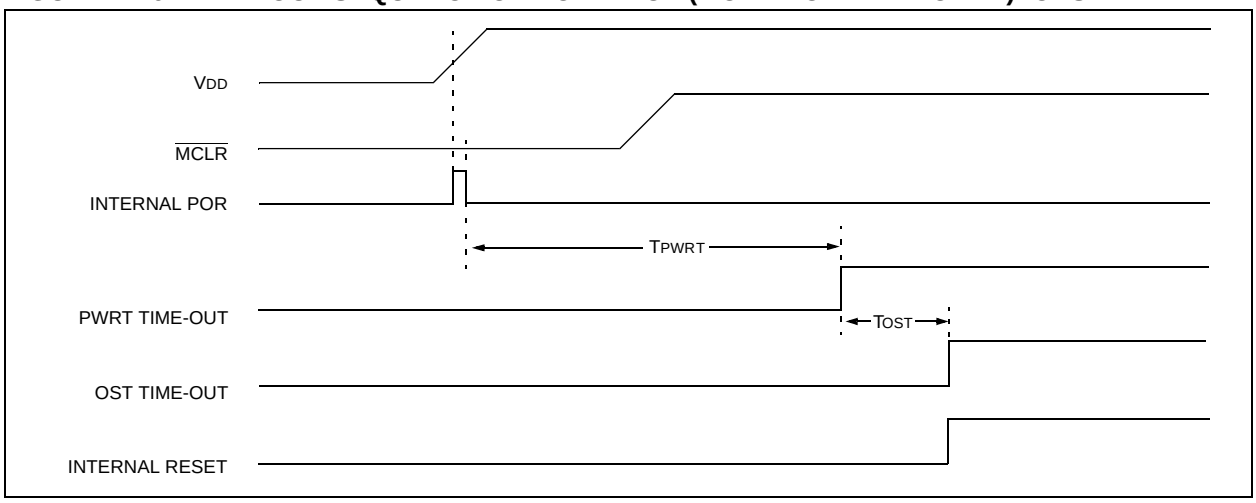


FIGURE 12-7: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ NOT TIED TO V_{DD}): CASE 2

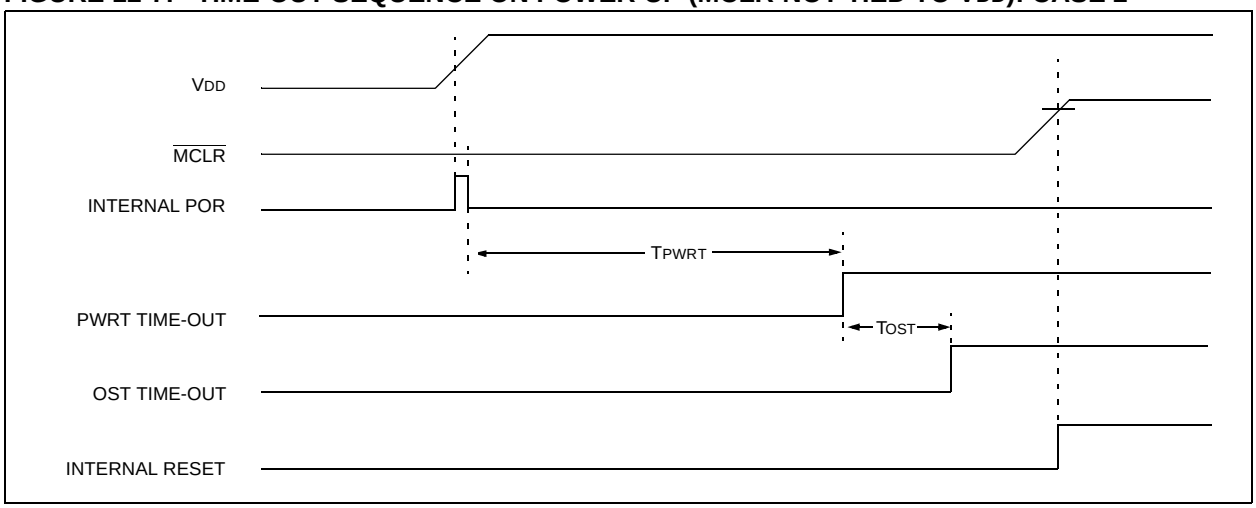
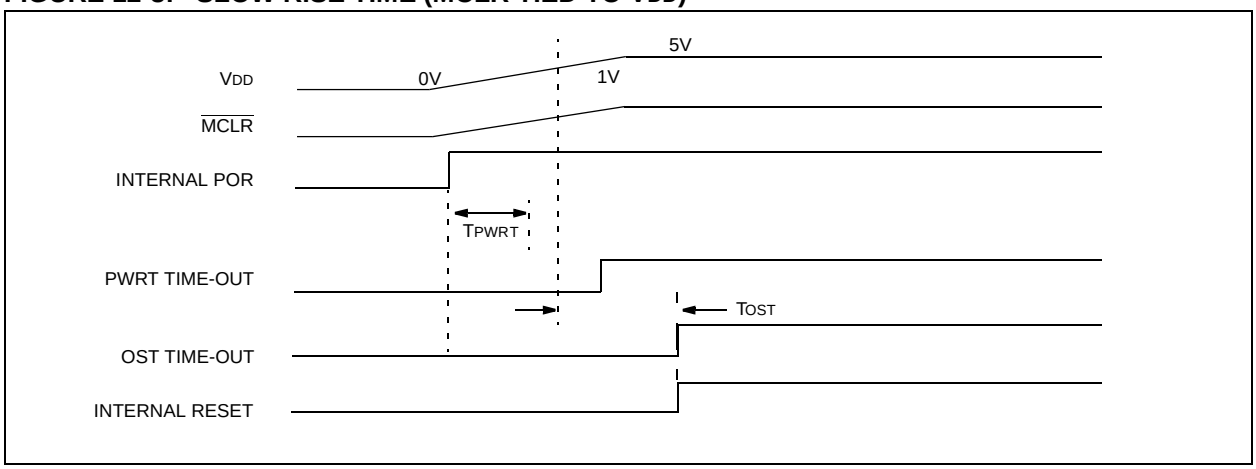


FIGURE 12-8: SLOW RISE TIME ($\overline{MCLR}$ TIED TO V_{DD})



© 1999 Microchip Technology Inc.

12.10 Interrupts

The PIC16F87X family has up to 14 sources of interrupt. The interrupt control register (INTCON) records individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

Note:	Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.

A global interrupt enable bit, GIE (INTCON<7>) enables (if set) all un-masked interrupts or disables (if cleared) all interrupts. When bit GIE is enabled, and an interrupt's flag bit and mask bit are set, the interrupt will vector immediately. Individual interrupts can be disabled through their corresponding enable bits in various registers. Individual interrupt bits are set regardless of the status of the GIE bit. The GIE bit is cleared on reset.

The "return from interrupt" instruction, `RETFIE`, exits the interrupt routine, as well as sets the GIE bit, which re-enables interrupts.

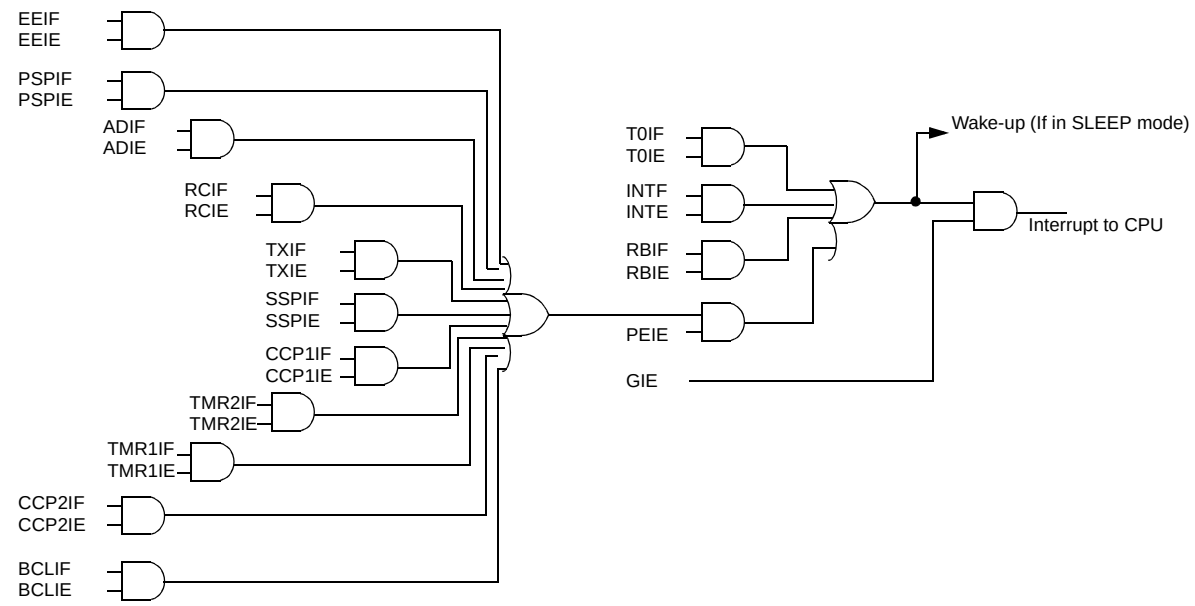
The RB0/INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

The peripheral interrupt flags are contained in the special function registers, PIR1 and PIR2. The corresponding interrupt enable bits are contained in special function registers, PIE1 and PIE2, and the peripheral interrupt enable bit is contained in special function register INTCON.

When an interrupt is responded to, the GIE bit is cleared to disable any further interrupt, the return address is pushed onto the stack and the PC is loaded with 0004h. Once in the interrupt service routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid recursive interrupts.

For external interrupt events, such as the INT pin or PORTB change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends when the interrupt event occurs. The latency is the same for one or two cycle instructions. Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit

FIGURE 12-9: INTERRUPT LOGIC



The following table shows which devices have which interrupts.

Device	T0IF	INTF	RBIF	PSPIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	EEIF	BCLIF	CCP2IF
PIC16F876/873	Yes	Yes	Yes	-	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PIC16F877/874	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

12.10.1 INT INTERRUPT

External interrupt on the RB0/INT pin is edge triggered, either rising, if bit INTEDG (OPTION_REG<6>) is set, or falling, if the INTEDG bit is clear. When a valid edge appears on the RB0/INT pin, flag bit INTF (INTCON<1>) is set. This interrupt can be disabled by clearing enable bit INTE (INTCON<4>). Flag bit INTF must be cleared in software in the interrupt service routine before re-enabling this interrupt. The INT interrupt can wake-up the processor from SLEEP, if bit INTE was set prior to going into SLEEP. The status of global interrupt enable bit GIE decides whether or not the processor branches to the interrupt vector following wake-up. See Section 12.13 for details on SLEEP mode.

12.10.2 TMR0 INTERRUPT

An overflow (FFh → 00h) in the TMR0 register will set flag bit T0IF (INTCON<2>). The interrupt can be enabled/disabled by setting/clearing enable bit T0IE (INTCON<5>). (Section 5.0)

12.10.3 PORTB INTCON CHANGE

An input change on PORTB<7:4> sets flag bit RBIF (INTCON<0>). The interrupt can be enabled/disabled by setting/clearing enable bit RBIE (INTCON<4>). (Section 3.2)

12.11 Context Saving During Interrupts

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt, (i.e., W register and STATUS register). This will have to be implemented in software.

For the PIC16F873/874 devices, the register W_TEMP must be defined in both banks 0 and 1 and must be defined at the same offset from the bank base address (i.e., If W_TEMP is defined at 0x20 in bank 0, it must also be defined at 0xA0 in bank 1.). The registers, PCLATH_TEMP and STATUS_TEMP, are only defined in bank 0.

Since the upper 16 bytes of each bank are common in the PIC16F876/877 devices, temporary holding registers W_TEMP, STATUS_TEMP and PCLATH_TEMP should be placed in here. These 16 locations don't require banking and therefore, make it easier for context save and restore. The same basic code in Example 12-1 can be used.

EXAMPLE 12-1: SAVING STATUS, W, AND PCLATH REGISTERS IN RAM

```
MOVWF    W_TEMP           ;Copy W to TEMP register
SWAPF    STATUS,W         ;Swap status to be saved into W
CLRF     STATUS           ;bank 0, regardless of current bank, Clears IRP,RP1,RP0
MOVWF    STATUS_TEMP      ;Save status to bank zero STATUS_TEMP register
MOVF     PCLATH, W        ;Only required if using pages 1, 2 and/or 3
MOVWF    PCLATH_TEMP      ;Save PCLATH into W
CLRF     PCLATH           ;Page zero, regardless of current page
:
:(ISR)
:
MOVF     PCLATH_TEMP, W   ;Restore PCLATH
MOVWF    PCLATH           ;Move W into PCLATH
SWAPF    STATUS_TEMP,W    ;Swap STATUS_TEMP register into W
                          ;(sets bank to original state)
MOVWF    STATUS           ;Move W into STATUS register
SWAPF    W_TEMP,F         ;Swap W_TEMP
SWAPF    W_TEMP,W         ;Swap W_TEMP into W
```


© 1999 Microchip Technology Inc.

12.12 Watchdog Timer (WDT)

The Watchdog Timer is as a free running on-chip RC oscillator which does not require any external components. This RC oscillator is separate from the RC oscillator of the OSC1/CLKIN pin. That means that the WDT will run, even if the clock on the OSC1/CLKIN and OSC2/CLKOUT pins of the device has been stopped, for example, by execution of a SLEEP instruction.

During normal operation, a WDT time-out generates a device RESET (Watchdog Timer Reset). If the device is in SLEEP mode, a WDT time-out causes the device to wake-up and continue with normal operation (Watchdog Timer Wake-up). The $\overline{TO}$ bit in the STATUS register will be cleared upon a Watchdog Timer time-out.

The WDT can be permanently disabled by clearing configuration bit WDTE (Section 12.1).

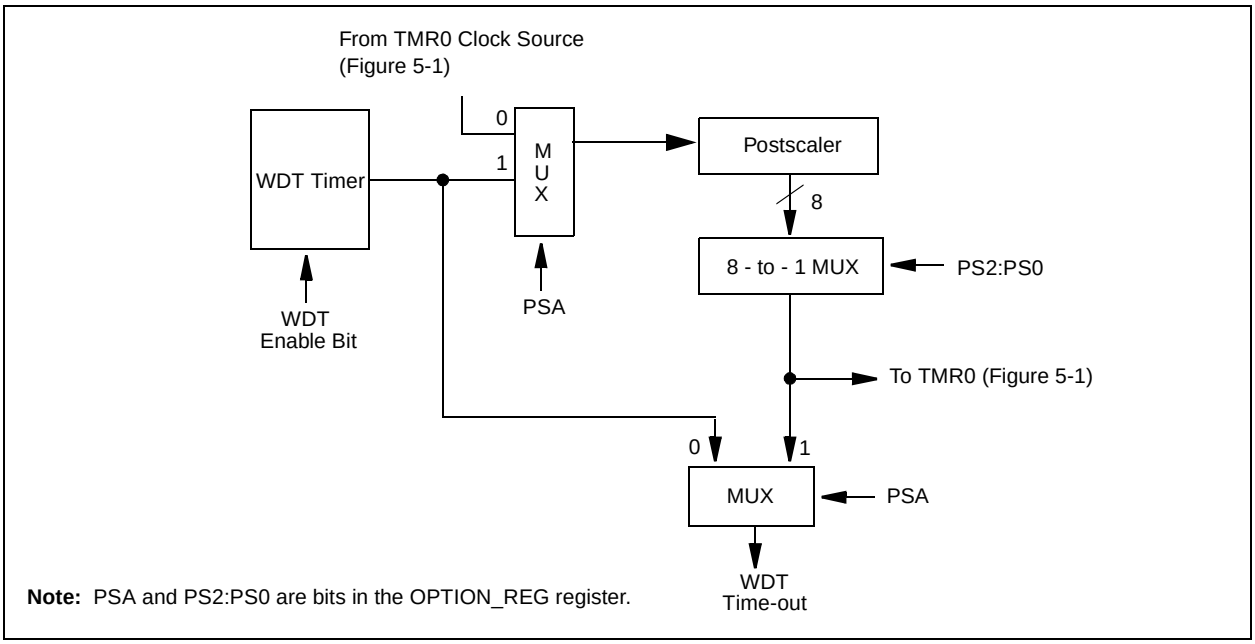
WDT time-out period values may be found in the Electrical Specifications section under parameter #31. Values for the WDT prescaler (actually a postscaler, but shared with the Timer0 prescaler) may be assigned using the OPTION_REG register.

Note:	The CLRWDT and SLEEP instructions clear the WDT and the postscaler, if assigned to the WDT, and prevent it from timing out and generating a device RESET condition.

.

Note:	When a CLRWDT instruction is executed and the prescaler is assigned to the WDT, the prescaler count will be cleared, but the prescaler assignment is not changed.

FIGURE 12-10: WATCHDOG TIMER BLOCK DIAGRAM



Note: PSA and PS2:PS0 are bits in the OPTION_REG register.

FIGURE 12-11: SUMMARY OF WATCHDOG TIMER REGISTERS

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2007h	Config. bits	**(1)**	BODEN(1)	CP1	CP0	$\overline{PWRTE}$(1)	WDTE	FOSC1	FOSC0
81h,181h	OPTION_REG	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

Legend: Shaded cells are not used by the Watchdog Timer.
Note 1: See Register 12-1 for operation of these bits.

12.13 Power-down Mode (SLEEP)

Power-down mode is entered by executing a SLEEP instruction.

If enabled, the Watchdog Timer will be cleared but keeps running, the $\overline{PD}$ bit (STATUS<3>) is cleared, the $\overline{TO}$ (STATUS<4>) bit is set, and the oscillator driver is turned off. The I/O ports maintain the status they had before the SLEEP instruction was executed (driving high, low, or hi-impedance).

For lowest current consumption in this mode, place all I/O pins at either V_{DD} or V_{SS}, ensure no external circuitry is drawing current from the I/O pin, power-down the A/D and disable external clocks. Pull all I/O pins that are hi-impedance inputs, high or low externally, to avoid switching currents caused by floating inputs. The T0CKI input should also be at V_{DD} or V_{SS} for lowest current consumption. The contribution from on-chip pull-ups on PORTB should be considered.

The $\overline{MCLR}$ pin must be at a logic high level (V_{IHMC}).

12.13.1 WAKE-UP FROM SLEEP

The device can wake up from SLEEP through one of the following events:

1. External reset input on $\overline{MCLR}$ pin.
2. Watchdog Timer wake-up (if WDT was enabled).
3. Interrupt from INT pin, RB port change or some Peripheral Interrupts.

External $\overline{MCLR}$ Reset will cause a device reset. All other events are considered a continuation of program execution and cause a "wake-up". The $\overline{TO}$ and $\overline{PD}$ bits in the STATUS register can be used to determine the cause of device reset. The $\overline{PD}$ bit, which is set on power-up, is cleared when SLEEP is invoked. The $\overline{TO}$ bit is cleared if a WDT time-out occurred and caused wake-up.

The following peripheral interrupts can wake the device from SLEEP:

1. PSP read or write.
2. TMR1 interrupt. Timer1 must be operating as an asynchronous counter.
3. CCP capture mode interrupt.
4. Special event trigger (Timer1 in asynchronous mode using an external clock).
5. SSP (Start/Stop) bit detect interrupt.
6. SSP transmit or receive in slave mode (SPI/I^2C).
7. USART RX or TX (synchronous slave mode).
8. A/D conversion (when A/D clock source is RC).
9. EEPROM write operation completion

Other peripherals cannot generate interrupts since during SLEEP, no on-chip clocks are present.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is pre-fetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the GIE bit. If the GIE bit is clear (disabled), the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is set (enabled), the device executes the instruction after the SLEEP instruction and then branches to the interrupt address (0004h). In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

12.13.2 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

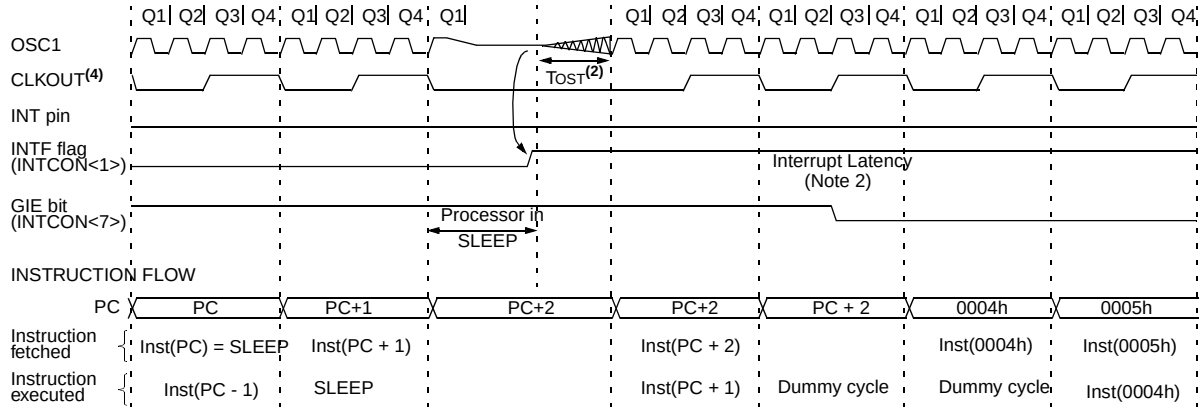
- If the interrupt occurs **before** the execution of a SLEEP instruction, the SLEEP instruction will complete as a NOP. Therefore, the WDT and WDT postscaler will not be cleared, the $\overline{TO}$ bit will not be set and $\overline{PD}$ bits will not be cleared.
- If the interrupt occurs **during or after** the execution of a SLEEP instruction, the device will immediately wake up from sleep. The SLEEP instruction will be completely executed before the wake-up. Therefore, the WDT and WDT postscaler will be cleared, the $\overline{TO}$ bit will be set and the $\overline{PD}$ bit will be cleared.

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the $\overline{PD}$ bit. If the $\overline{PD}$ bit is set, the SLEEP instruction was executed as a NOP.

To ensure that the WDT is cleared, a CLRWDT instruction should be executed before a SLEEP instruction.

© 1999 Microchip Technology Inc.

FIGURE 12-12: WAKE-UP FROM SLEEP THROUGH INTERRUPT



Note 1: XT, HS or LP oscillator mode assumed.
2: TOST = 1024TOSC (drawing not to scale) This delay will not be there for RC osc mode.
3: GIE = '1' assumed. In this case after wake- up, the processor jumps to the interrupt routine. If GIE = '0', execution will continue in-line.
4: CLKOUT is not available in these osc modes, but shown here for timing reference.

12.14 In-Circuit Debugger

When the DEBUG bit in the configuration word is programmed to a '0', the In-Circuit Debugger functionality is enabled. This function allows simple debugging functions when used with MPLAB. When the microcontroller has this feature enabled, some of the resources are not available for general use. Table 12-7 shows which features are consumed by the background debugger.

TABLE 12-7: DEBUGGER RESOURCES

I/O pins	RB6, RB7
Stack	1 level
Program Memory	Address 0000h must be NOP
	Last 100h words
Data Memory	0x070(0x0F0, 0x170, 0x1F0) 0x1EB - 0x1EF

To use the In-Circuit Debugger function of the microcontroller, the design must implement In-Circuit Serial Programming connections to $\overline{MCLR}$/VPP, VDD, GND, RB7 and RB6. This will interface to the In-Circuit Debugger module available from Microchip or one of the third party development tool companies.

12.15 Program Verification/Code Protection

If the code protection bit(s) have not been programmed, the on-chip program memory can be read out for verification purposes.

12.16 ID Locations

Four memory locations (2000h - 2003h) are designated as ID locations where the user can store checksum or other code-identification numbers. These locations are not accessible during normal execution but are readable and writable during program/verify. It is recommended that only the 4 least significant bits of the ID location are used.

12.17 In-Circuit Serial Programming

PIC16F87X microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data and three other lines for power, ground, and the programming voltage. This allows customers to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

When using ICSP, the part must be supplied 4.5V to 5.5V if a bulk erase will be executed. This includes reprogramming of the code protect both from an on-state to off-state. For all other cases of ICSP, the part may be programmed at the normal operating voltages. This means calibration values, unique user IDs or user code can be reprogrammed or added.

For complete details of serial programming, please refer to the In-Circuit Serial Programming (ICSP™) Guide, (DS30277B).

12.18 Low Voltage ICSP Programming

The LVP bit of the configuration word enables low voltage ICSP programming. This mode allows the microcontroller to be programmed via ICSP using a V_{DD} source in the operating voltage range. This only means that V_{PP} does not have to be brought to V_{IHH}, but can instead be left at the normal operating voltage. In this mode, the RB3/PGM pin is dedicated to the programming function and ceases to be a general purpose I/O pin. During programming, V_{DD} is applied to the $\overline{MCLR}$ pin. To enter programming mode, V_{DD} must be applied to the RB3/PGM provided the LVP bit is set. The LVP bit defaults to on (‘1’) from the factory.

Note 1: The high voltage programming mode is always available, regardless of the state of the LVP bit, by applying V_{IHH} to the $\overline{MCLR}$ pin.

2: While in low voltage ICSP mode, the RB3 pin can no longer be used as a general purpose I/O pin.

3: When using low voltage ICSP programming (LVP) and the pull-ups on PORTB are enabled, bit 3 in the TRISB register must be cleared to disable the pull-up on RB3 and ensure the proper operation of the device.

If low-voltage programming mode is not used, the LVP bit can be programmed to a '0' and RB3/PGM becomes a digital I/O pin. However, the LVP bit may only be programmed when programming is entered with V_{IHH} on $\overline{MCLR}$. The LVP bit can only be charged when using high voltage on $\overline{MCLR}$.

It should be noted, that once the LVP bit is programmed to 0, only the high voltage programming mode is available and only high voltage programming mode can be used to program the device.

When using low voltage ICSP, the part must be supplied 4.5V to 5.5V if a bulk erase will be executed. This includes reprogramming of the code protect bits from an on-state to off-state. For all other cases of low voltage ICSP, the part may be programmed at the normal operating voltage. This means calibration values, unique user IDs or user code can be reprogrammed or added.

© 1999 Microchip Technology Inc.

13.0 INSTRUCTION SET SUMMARY

Each PIC16CXX instruction is a 14-bit word divided into an OPCODE which specifies the instruction type and one or more operands which further specify the operation of the instruction. The PIC16CXX instruction set summary in Table 13-2 lists **byte-oriented**, **bit-oriented**, and **literal and control** operations. Table 13-1 shows the opcode field descriptions.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, the result is placed in the file register specified in the instruction.

For **bit-oriented** instructions, 'b' represents a bit field designator which selects the number of the bit affected by the operation, while 'f' represents the number of the file in which the bit is located.

For **literal and control** operations, 'k' represents an eight or eleven bit constant or literal value.

TABLE 13-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1) The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1
PC	Program Counter
TO	Time-out bit
PD	Power-down bit

The instruction set is highly orthogonal and is grouped into three basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

All instructions are executed within one single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles with the second cycle executed as a NOP. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 μs. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 μs.

Table 13-2 lists the instructions recognized by the MPASM assembler.

Figure 13-1 shows the general formats that the instructions can have.

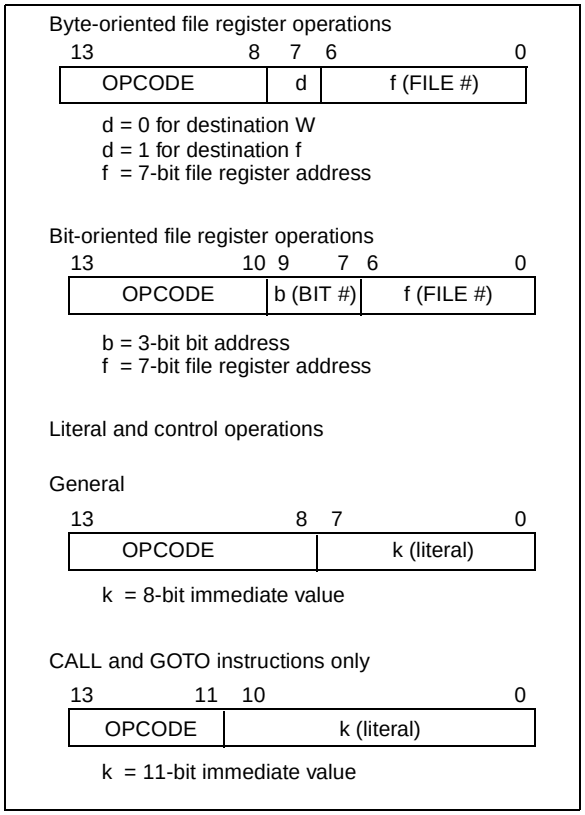
Note:	To maintain upward compatibility with future PIC16CXX products, do not use the OPTION and TRIS instructions.

All examples use the following format to represent a hexadecimal number:

0xhh

where h signifies a hexadecimal digit.

FIGURE 13-1: GENERAL FORMAT FOR INSTRUCTIONS



A description of each instruction is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

TABLE 13-2: PIC16CXXX INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS						
ADDWF	f, d	Add W and f	1	00 0111 dfff ffff	C,DC,Z	1,2
ANDWF	f, d	AND W with f	1	00 0101 dfff ffff	Z	1,2
CLRF	f	Clear f	1	00 0001 lfff ffff	Z	2
CLRW	-	Clear W	1	00 0001 0xxx xxxx	Z	
COMF	f, d	Complement f	1	00 1001 dfff ffff	Z	1,2
DECF	f, d	Decrement f	1	00 0011 dfff ffff	Z	1,2
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00 1011 dfff ffff		1,2,3
INCF	f, d	Increment f	1	00 1010 dfff ffff	Z	1,2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00 1111 dfff ffff		1,2,3
IORWF	f, d	Inclusive OR W with f	1	00 0100 dfff ffff	Z	1,2
MOVF	f, d	Move f	1	00 1000 dfff ffff	Z	1,2
MOVWF	f	Move W to f	1	00 0000 lfff ffff		
NOP	-	No Operation	1	00 0000 0xx0 0000		
RLF	f, d	Rotate Left f through Carry	1	00 1101 dfff ffff	C	1,2
RRF	f, d	Rotate Right f through Carry	1	00 1100 dfff ffff	C	1,2
SUBWF	f, d	Subtract W from f	1	00 0010 dfff ffff	C,DC,Z	1,2
SWAPF	f, d	Swap nibbles in f	1	00 1110 dfff ffff		1,2
XORWF	f, d	Exclusive OR W with f	1	00 0110 dfff ffff	Z	1,2
BIT-ORIENTED FILE REGISTER OPERATIONS						
BCF	f, b	Bit Clear f	1	01 00bb bfff ffff		1,2
BSF	f, b	Bit Set f	1	01 01bb bfff ffff		1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01 10bb bfff ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01 11bb bfff ffff		3
LITERAL AND CONTROL OPERATIONS						
ADDLW	k	Add literal and W	1	11 111x kkkk kkkk	C,DC,Z	
ANDLW	k	AND literal with W	1	11 1001 kkkk kkkk	Z	
CALL	k	Call subroutine	2	10 0kkk kkkk kkkk		
CLRWDT	-	Clear Watchdog Timer	1	00 0000 0110 0100	$\overline{TO}$,$\overline{PD}$	
GOTO	k	Go to address	2	10 1kkk kkkk kkkk		
IORLW	k	Inclusive OR literal with W	1	11 1000 kkkk kkkk	Z	
MOVLW	k	Move literal to W	1	11 00xx kkkk kkkk		
RETFIE	-	Return from interrupt	2	00 0000 0000 1001		
RETLW	k	Return with literal in W	2	11 01xx kkkk kkkk		
RETURN	-	Return from Subroutine	2	00 0000 0000 1000		
SLEEP	-	Go into standby mode	1	00 0000 0110 0011	$\overline{TO}$,$\overline{PD}$	
SUBLW	k	Subtract W from literal	1	11 110x kkkk kkkk	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11 1010 kkkk kkkk	Z	

Note 1: When an I/O register is modified as a function of itself (e.g., `MOVF PORTB, 1`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 Module.

3: If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a `NOP`.

Note: Additional information on the mid-range instruction set is available in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).

© 1999 Microchip Technology Inc.

13.1 Instruction Descriptions

ADDLW	Add Literal and W
Syntax:	[*label*] ADDLW k
Operands:	$0 \le k \le 255$
Operation:	(W) + k → (W)
Status Affected:	C, DC, Z
Description:	The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register.

ADDWF	Add W and f
Syntax:	[*label*] ADDWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) + (f) → (destination)
Status Affected:	C, DC, Z
Description:	Add the contents of the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

ANDLW	AND Literal with W
Syntax:	[*label*] ANDLW k
Operands:	$0 \le k \le 255$
Operation:	(W) .AND. (k) → (W)
Status Affected:	Z
Description:	The contents of W register are AND'ed with the eight bit literal 'k'. The result is placed in the W register.

ANDWF	AND W with f
Syntax:	[*label*] ANDWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) .AND. (f) → (destination)
Status Affected:	Z
Description:	AND the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

BCF	Bit Clear f
Syntax:	[*label*] BCF f,b
Operands:	$0 \le f \le 127$ $0 \le b \le 7$
Operation:	0 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is cleared.

BSF	Bit Set f
Syntax:	[*label*] BSF f,b
Operands:	$0 \le f \le 127$ $0 \le b \le 7$
Operation:	1 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is set.

BTFSS	Bit Test f, Skip if Set
Syntax:	[*label*] BTFSS f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b < 7
Operation:	skip if (f<b>) = 1
Status Affected:	None
Description:	If bit 'b' in register 'f' is '0', the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead making this a 2TCY instruction.

BTFSC	Bit Test, Skip if Clear
Syntax:	[*label*] BTFSC f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	skip if (f<b>) = 0
Status Affected:	None
Description:	If bit 'b' in register 'f' is '1', the next instruction is executed. If bit 'b', in register 'f', is '0', the next instruction is discarded, and a NOP is executed instead, making this a 2TCY instruction.

CALL	Call Subroutine
Syntax:	[*label*] CALL k
Operands:	0 ≤ k ≤ 2047
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<4:3>) → PC<12:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two cycle instruction.

CLRF	Clear f
Syntax:	[*label*] CLRF f
Operands:	0 ≤ f ≤ 127
Operation:	00h → (f) 1 → Z
Status Affected:	Z
Description:	The contents of register 'f' are cleared and the Z bit is set.

CLRW	Clear W
Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h → (W) 1 → Z
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

CLRWDT	Clear Watchdog Timer
Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h → WDT 0 → WDT prescaler, 1 → $\overline{TO}$ 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

© 1999 Microchip Technology Inc.

COMF	Complement f
Syntax:	[*label*] COMF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	$(\bar{f}) \rightarrow$ (destination)
Status Affected:	Z
Description:	The contents of register 'f' are complemented. If 'd' is 0, the result is stored in W. If 'd' is 1, the result is stored back in register 'f'.

DECF	Decrement f
Syntax:	[*label*] DECF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - 1 $\rightarrow$ (destination)
Status Affected:	Z
Description:	Decrement register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

DECFSZ	Decrement f, Skip if 0
Syntax:	[*label*] DECFSZ f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - 1 $\rightarrow$ (destination); skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are decremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'. If the result is 1, the next instruction is executed. If the result is 0, then a NOP is executed instead making it a 2TCY instruction.

GOTO	Unconditional Branch
Syntax:	[*label*] GOTO k
Operands:	$0 \le k \le 2047$
Operation:	k $\rightarrow$ PC<10:0> PCLATH<4:3> $\rightarrow$ PC<12:11>
Status Affected:	None
Description:	GOTO is an unconditional branch. The eleven bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two cycle instruction.

INCF	Increment f
Syntax:	[*label*] INCF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) + 1 $\rightarrow$ (destination)
Status Affected:	Z
Description:	The contents of register 'f' are incremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.

INCFSZ	Increment f, Skip if 0
Syntax:	[*label*] INCFSZ f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) + 1 $\rightarrow$ (destination), skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are incremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'. If the result is 1, the next instruction is executed. If the result is 0, a NOP is executed instead making it a 2TCY instruction.

IORLW	Inclusive OR Literal with W
Syntax:	[*label*] IORLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) .OR. k → (W)
Status Affected:	Z
Description:	The contents of the W register are OR'ed with the eight bit literal 'k'. The result is placed in the W register.

IORWF	Inclusive OR W with f
Syntax:	[*label*] IORWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) .OR. (f) → (destination)
Status Affected:	Z
Description:	Inclusive OR the W register with register 'f'. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.

MOVF	Move f
Syntax:	[*label*] MOVF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f) → (destination)
Status Affected:	Z
Description:	The contents of register f are moved to a destination dependant upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.

MOVLW	Move Literal to W
Syntax:	[*label*] MOVLW k
Operands:	0 ≤ k ≤ 255
Operation:	k → (W)
Status Affected:	None
Description:	The eight bit literal 'k' is loaded into W register. The don't cares will assemble as 0's.

MOVWF	Move W to f
Syntax:	[*label*] MOVWF f
Operands:	0 ≤ f ≤ 127
Operation:	(W) → (f)
Status Affected:	None
Description:	Move data from W register to register 'f'.

NOP	No Operation
Syntax:	[*label*] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.

© 1999 Microchip Technology Inc.

RETFIE **Return from Interrupt**

Syntax: [*label*] RETFIE

Operands: None

Operation: TOS → PC,
1 → GIE

Status Affected: None

RETLW **Return with Literal in W**

Syntax: [*label*] RETLW k

Operands: 0 ≤ k ≤ 255

Operation: k → (W);
TOS → PC

Status Affected: None

Description: The W register is loaded with the eight bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two cycle instruction.

RETURN **Return from Subroutine**

Syntax: [*label*] RETURN

Operands: None

Operation: TOS → PC

Status Affected: None

Description: Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two cycle instruction.

RLF **Rotate Left f through Carry**

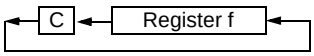
Syntax: [*label*] RLF f,d

Operands: 0 ≤ f ≤ 127
d ∈ [0,1]

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the left through the Carry Flag. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is stored back in register 'f'.



RRF **Rotate Right f through Carry**

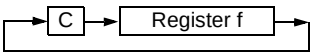
Syntax: [*label*] RRF f,d

Operands: 0 ≤ f ≤ 127
d ∈ [0,1]

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the right through the Carry Flag. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.



SLEEP

Syntax: [*label*] SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT prescaler,
1 → $\overline{TO}$,
0 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Description: The power-down status bit, $\overline{PD}$ is cleared. Time-out status bit, $\overline{TO}$ is set. Watchdog Timer and its prescaler are cleared.
The processor is put into SLEEP mode with the oscillator stopped.

SUBLW **Subtract W from Literal**

Syntax:	[*label*] SUBLW k
Operands:	$0 \le k \le 255$
Operation:	k - (W) → (W)
Status Affected:	C, DC, Z
Description:	The W register is subtracted (2's complement method) from the eight bit literal 'k'. The result is placed in the W register.

SUBWF **Subtract W from f**

Syntax:	[*label*] SUBWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - (W) → (destination)
Status Affected:	C, DC, Z
Description:	Subtract (2's complement method) W register from register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

SWAPF **Swap Nibbles in f**

Syntax:	[*label*] SWAPF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f<3:0>) → (destination<7:4>), (f<7:4>) → (destination<3:0>)
Status Affected:	None
Description:	The upper and lower nibbles of register 'f' are exchanged. If 'd' is 0, the result is placed in W register. If 'd' is 1, the result is placed in register 'f'.

XORLW **Exclusive OR Literal with W**

Syntax:	[*label*] XORLW k
Operands:	$0 \le k \le 255$
Operation:	(W) .XOR. k → (W)
Status Affected:	Z
Description:	The contents of the W register are XOR'ed with the eight bit literal 'k'. The result is placed in the W register.

XORWF **Exclusive OR W with f**

Syntax:	[*label*] XORWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) .XOR. (f) → (destination)
Status Affected:	Z
Description:	Exclusive OR the contents of the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

© 1999 Microchip Technology Inc.

14.0 DEVELOPMENT SUPPORT

The PICmicro® microcontrollers are supported with a full range of hardware and software development tools:

- Integrated Development Environment
  - MPLAB® IDE Software
- Assemblers/Compilers/Linkers
  - MPASM Assembler
  - MPLAB-C17 and MPLAB-C18 C Compilers
  - MPLINK/MPLIB Linker/Librarian
- Simulators
  - MPLAB-SIM Software Simulator
- Emulators
  - MPLAB-ICE Real-Time In-Circuit Emulator
  - PICMASTER®/PICMASTER-CE In-Circuit Emulator
  - ICEPIC™
- In-Circuit Debugger
  - MPLAB-ICD for PIC16F877
- Device Programmers
  - PRO MATE® II Universal Programmer
  - PICSTART® Plus Entry-Level Prototype Programmer
- Low-Cost Demonstration Boards
  - SIMICE
  - PICDEM-1
  - PICDEM-2
  - PICDEM-3
  - PICDEM-17
  - SEEVAL®
  - KEELOQ®

14.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8-bit microcontroller market. MPLAB is a Windows®-based application which contains:

- Multiple functionality
  - editor
  - simulator
  - programmer (sold separately)
  - emulator (sold separately)
- A full featured editor
- A project manager
- Customizable tool bar and key mapping
- A status bar
- On-line help

MPLAB allows you to:

- Edit your source files (either assembly or 'C')
- One touch assemble (or compile) and download to PICmicro tools (automatically updates all project information)
- Debug using:
  - source files
  - absolute listing file
  - object code

The ability to use MPLAB with Microchip's simulator, MPLAB-SIM, allows a consistent platform and the ability to easily switch from the cost-effective simulator to the full featured emulator with minimal retraining.

14.2 MPASM Assembler

MPASM is a full featured universal macro assembler for all PICmicro MCU's. It can produce absolute code directly in the form of HEX files for device programmers, or it can generate relocatable objects for MPLINK.

MPASM has a command line interface and a Windows shell and can be used as a standalone application on a Windows 3.x or greater system. MPASM generates relocatable object files, Intel standard HEX files, MAP files to detail memory usage and symbol reference, an absolute LST file which contains source lines and generated machine code, and a COD file for MPLAB debugging.

MPASM features include:

- MPASM and MPLINK are integrated into MPLAB projects.
- MPASM allows user defined macros to be created for streamlined assembly.
- MPASM allows conditional assembly for multi purpose source files.
- MPASM directives allow complete control over the assembly process.

14.3 MPLAB-C17 and MPLAB-C18 C Compilers

The MPLAB-C17 and MPLAB-C18 Code Development Systems are complete ANSI 'C' compilers and integrated development environments for Microchip's PIC17CXXX and PIC18CXXX family of microcontrollers, respectively. These compilers provide powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compilers provide symbol information that is compatible with the MPLAB IDE memory display.

14.4 MPLINK/MPLIB Linker/Librarian

MPLINK is a relocatable linker for MPASM and MPLAB-C17 and MPLAB-C18. It can link relocatable objects from assembly or C source files along with pre-compiled libraries using directives from a linker script.

MPLIB is a librarian for pre-compiled code to be used with MPLINK. When a routine from a library is called from another source file, only the modules that contains that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications. MPLIB manages the creation and modification of library files.

MPLINK features include:

- MPLINK works with MPASM and MPLAB-C17 and MPLAB-C18.
- MPLINK allows all memory areas to be defined as sections to provide link-time flexibility.

MPLIB features include:

- MPLIB makes linking easier because single libraries can be included instead of many smaller files.
- MPLIB helps keep code maintainable by grouping related modules together.
- MPLIB commands allow libraries to be created and modules to be added, listed, replaced, deleted, or extracted.

14.5 MPLAB-SIM Software Simulator

The MPLAB-SIM Software Simulator allows code development in a PC host environment by simulating the PICmicro series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file or user-defined key press to any of the pins. The execution can be performed in single step, execute until break, or trace mode.

MPLAB-SIM fully supports symbolic debugging using MPLAB-C17 and MPLAB-C18 and MPASM. The Software Simulator offers the flexibility to develop and debug code outside of the laboratory environment making it an excellent multi-project software development tool.

14.6 MPLAB-ICE High Performance Universal In-Circuit Emulator with MPLAB IDE

The MPLAB-ICE Universal In-Circuit Emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PICmicro microcontrollers (MCUs). Software control of MPLAB-ICE is provided by the MPLAB Integrated Development Environment (IDE), which allows editing, “make” and download, and source debugging from a single environment.

Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the MPLAB-ICE allows expansion to support new PICmicro microcontrollers.

The MPLAB-ICE Emulator System has been designed as a real-time emulation system with advanced features that are generally found on more expensive development tools. The PC platform and Microsoft® Windows 3.x/95/98 environment were chosen to best make these features available to you, the end user.

MPLAB-ICE 2000 is a full-featured emulator system with enhanced trace, trigger, and data monitoring features. Both systems use the same processor modules and will operate across the full operating speed range of the PICmicro MCU.

14.7 PICMASTER/PICMASTER CE

The PICMASTER system from Microchip Technology is a full-featured, professional quality emulator system. This flexible in-circuit emulator provides a high-quality, universal platform for emulating Microchip 8-bit PICmicro microcontrollers (MCUs). PICMASTER systems are sold worldwide, with a CE compliant model available for European Union (EU) countries.

14.8 ICEPIC

ICEPIC is a low-cost in-circuit emulation solution for the Microchip Technology PIC16C5X, PIC16C6X, PIC16C7X, and PIC16CXXX families of 8-bit one-time-programmable (OTP) microcontrollers. The modular system can support different subsets of PIC16C5X or PIC16CXXX products through the use of interchangeable personality modules or daughter boards. The emulator is capable of emulating without target application circuitry being present.

14.9 MPLAB-ICD In-Circuit Debugger

Microchip’s In-Circuit Debugger, MPLAB-ICD, is a powerful, low-cost run-time development tool. This tool is based on the flash PIC16F877 and can be used to develop for this and other PICmicro microcontrollers from the PIC16CXXX family. MPLAB-ICD utilizes the In-Circuit Debugging capability built into the PIC16F87X. This feature, along with Microchip’s In-Circuit Serial Programming protocol, offers cost-effective in-circuit flash programming and debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by watching variables, single-stepping and setting break points. Running at full speed enables testing hardware in real-time. The MPLAB-ICD is also a programmer for the flash PIC16F87X family.

© 1999 Microchip Technology Inc.

14.10 PRO MATE II Universal Programmer

The PRO MATE II Universal Programmer is a full-featured programmer capable of operating in stand-alone mode as well as PC-hosted mode. PRO MATE II is CE compliant.

The PRO MATE II has programmable V_{DD} and V_{PP} supplies which allows it to verify programmed memory at V_{DD} min and V_{DD} max for maximum reliability. It has an LCD display for instructions and error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode the PRO MATE II can read, verify or program PICmicro devices. It can also set code-protect bits in this mode.

14.11 PICSTART Plus Entry Level Development System

The PICSTART programmer is an easy-to-use, low-cost prototype programmer. It connects to the PC via one of the COM (RS-232) ports. MPLAB Integrated Development Environment software makes using the programmer simple and efficient.

PICSTART Plus supports all PICmicro devices with up to 40 pins. Larger pin count devices such as the PIC16C92X, and PIC17C76X may be supported with an adapter socket. PICSTART Plus is CE compliant.

14.12 SIMICE Entry-Level Hardware Simulator

SIMICE is an entry-level hardware development system designed to operate in a PC-based environment with Microchip's simulator MPLAB-SIM. Both SIMICE and MPLAB-SIM run under Microchip Technology's MPLAB Integrated Development Environment (IDE) software. Specifically, SIMICE provides hardware simulation for Microchip's PIC12C5XX, PIC12CE5XX, and PIC16C5X families of PICmicro 8-bit microcontrollers. SIMICE works in conjunction with MPLAB-SIM to provide non-real-time I/O port emulation. SIMICE enables a developer to run simulator code for driving the target system. In addition, the target system can provide input to the simulator code. This capability allows for simple and interactive debugging without having to manually generate MPLAB-SIM stimulus files. SIMICE is a valuable debugging tool for entry-level system development.

14.13 PICDEM-1 Low-Cost PICmicro Demonstration Board

The PICDEM-1 is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The users can program the sample microcontrollers provided with the PICDEM-1 board, on a PRO MATE II or PICSTART-Plus programmer, and easily test firmware. The user can also connect the PICDEM-1 board to the MPLAB-ICE emulator and download the firmware to the emulator for testing. Additional prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push-button switches and eight LEDs connected to PORTB.

14.14 PICDEM-2 Low-Cost PIC16CXX Demonstration Board

The PICDEM-2 is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-2 board, on a PRO MATE II programmer or PICSTART-Plus, and easily test firmware. The MPLAB-ICE emulator may also be used with the PICDEM-2 board to test firmware. Additional prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push-button switches, a potentiometer for simulated analog input, a Serial EEPROM to demonstrate usage of the I^2C bus and separate headers for connection to an LCD module and a keypad.

14.15 PICDEM-3 Low-Cost PIC16CXXX Demonstration Board

The PICDEM-3 is a simple demonstration board that supports the PIC16C923 and PIC16C924 in the PLCC package. It will also support future 44-pin PLCC microcontrollers with a LCD Module. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-3 board, on a PRO MATE II programmer or PICSTART Plus with an adapter socket, and easily test firmware. The MPLAB-ICE emulator may also be used with the PICDEM-3 board to test firmware. Additional prototype area has been provided to the user for adding hardware and connecting it to the microcontroller socket(s). Some of the features include an RS-232 interface, push-button switches, a potentiometer for simulated analog input, a thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM-3 board is an LCD panel, with 4 commons and 12 segments, that is capable of displaying time, temperature and day of the week. The PICDEM-3 provides an additional RS-232 interface and Windows 3.1 software for showing the demultiplexed LCD signals on a PC. A simple serial interface allows the user to construct a hardware demultiplexer for the LCD signals.

© 1999 Microchip Technology Inc.

14.16 PICDEM-17

The PICDEM-17 is an evaluation board that demonstrates the capabilities of several Microchip microcontrollers, including PIC17C752, PIC17C756, PIC17C762, and PIC17C766. All necessary hardware is included to run basic demo programs, which are supplied on a 3.5-inch disk. A programmed sample is included, and the user may erase it and program it with the other sample programs using the PRO MATE II or PICSTART Plus device programmers and easily debug and test the sample code. In addition, PICDEM-17 supports down-loading of programs to and executing out of external FLASH memory on board. The PICDEM-17 is also usable with the MPLAB-ICE or PICMASTER emulator, and all of the sample programs can be run and modified using either emulator. Additionally, a generous prototype area is available for user hardware.

14.17 SEEVAL Evaluation and Programming System

The SEEVAL SEEPROM Designer's Kit supports all Microchip 2-wire and 3-wire Serial EEPROMs. The kit includes everything necessary to read, write, erase or program special features of any Microchip SEEPROM product including Smart Serials™ and secure serials. The Total Endurance™ Disk is included to aid in trade-off analysis and reliability calculations. The total kit can significantly reduce time-to-market and result in an optimized system.

14.18 KEELOQ Evaluation and Programming Tools

KEELOQ evaluation and programming tools support Microchips HCS Secure Data Products. The HCS evaluation kit includes an LCD display to show changing codes, a decoder to decode transmissions, and a programming interface to program test transmitters.

© 1999 Microchip Technology Inc.

TABLE 14-1: DEVELOPMENT TOOLS FROM MICROCHIP

		PIC12CXXX	PIC14000	PIC16C5X	PIC16C6X	PIC16CXXX	PIC16F62X	PIC16C7X	PIC16C7XX	PIC16C8X	PIC16F8XX	PIC16C9XX	PIC17C4X	PIC17C7XX	PIC18CXX2	24CXX/ 25CXX/ 93CXX	HCSXXX	MCRFXXX	MCP2510
Software Tools	MPLAB® Integrated Development Environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
	MPLAB® C17 Compiler												✓	✓					
	MPLAB® C18 Compiler														✓				
	MPASM/MPLINK	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Emulators	MPLAB®-ICE	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	PICMASTER/PICMASTER-CE	✓	✓	✓	✓	✓		✓	✓	✓		✓	✓	✓					
	ICEPIC™ Low-Cost In-Circuit Emulator	✓		✓	✓	✓		✓	✓	✓		✓							
Debugger	MPLAB®-ICD In-Circuit Debugger				✓*			✓*			✓								
Programmers	PICSTART®Plus Low-Cost Universal Dev. Kit	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	PRO MATE® II Universal Programmer	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Demo Boards and Eval Kits	SIMICE	✓		✓															
	PICDEM-1			✓		✓		✓†		✓			✓						
	PICDEM-2				✓†			✓†							✓				
	PICDEM-3											✓							
	PICDEM-14A		✓																
	PICDEM-17													✓					
	KEELOQ® Evaluation Kit																✓		
	KEELOQ Transponder Kit																✓		
	microID™ Programmer's Kit																	✓	
	125 kHz microID Developer's Kit																	✓	
	125 kHz Anticollision microID Developer's Kit																	✓	
	13.56 MHz Anticollision microID Developer's Kit																	✓	
	MCP2510 CAN Developer's Kit																		✓

* Contact the Microchip Technology Inc. web site at www.microchip.com for information on how to use the MPLAB-ICD In-Circuit Debugger (DV164001) with PIC16C62, 63, 64, 65, 72, 73, 74, 76, 77

** Contact Microchip Technology Inc. for availability date.

† Development tool is available on select devices.

NOTES:

© 1999 Microchip Technology Inc.

15.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings †

Ambient temperature under bias ..... -55 to +125°C
Storage temperature ..... -65°C to +150°C
Voltage on any pin with respect to V_{SS} (except V_{DD}, $\overline{MCLR}$. and RA4) ..... -0.3V to (V_{DD} + 0.3V)
Voltage on V_{DD} with respect to V_{SS} ..... -0.3 to +7.5V
Voltage on $\overline{MCLR}$ with respect to V_{SS} (Note 2) ..... 0 to +14V
Voltage on RA4 with respect to Vss ..... 0 to +8.5V
Total power dissipation (Note 1) 1.0W
Maximum current out of V_{SS} pin ..... 300 mA
Maximum current into V_{DD} pin ..... 250 mA
Input clamp current, I_{IK} (V_I < 0 or V_I > V_{DD}) ..... ± 20 mA
Output clamp current, I_{OK} (V_O < 0 or V_O > V_{DD}) ..... ± 20 mA
Maximum output current sunk by any I/O pin ..... 25 mA
Maximum output current sourced by any I/O pin ..... 25 mA
Maximum current sunk by PORTA, PORTB, and PORTE (combined) (Note 3) ..... 200 mA
Maximum current sourced by PORTA, PORTB, and PORTE (combined) (Note 3) ..... 200 mA
Maximum current sunk by PORTC and PORTD (combined) (Note 3) ..... 200 mA
Maximum current sourced by PORTC and PORTD (combined) (Note 3) ..... 200 mA

Note 1: Power dissipation is calculated as follows: $Pdis = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum(Vol \times I_{OL})$

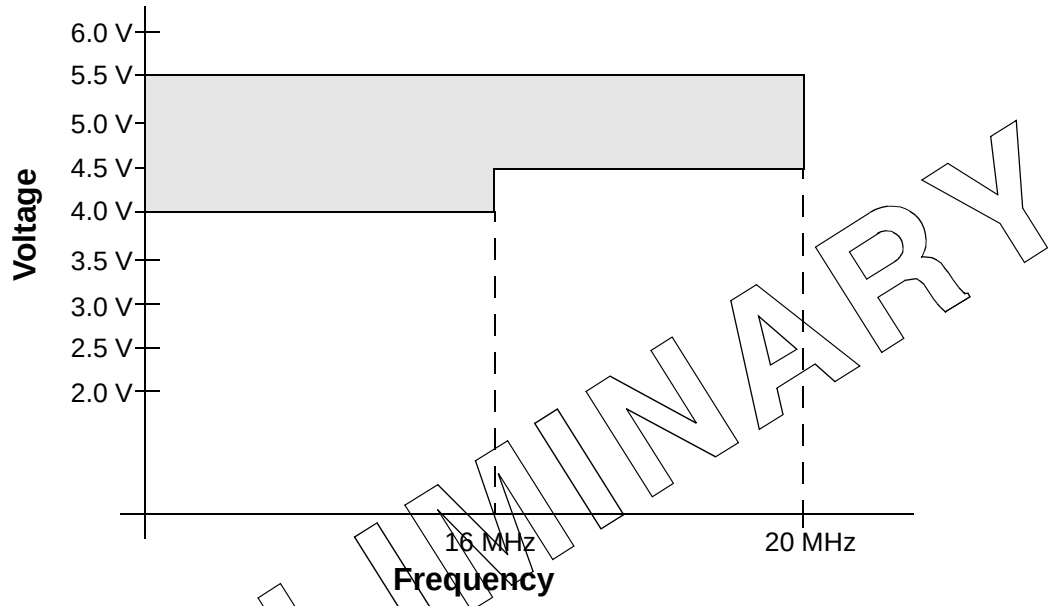
2: Voltage spikes below V_{SS} at the $\overline{MCLR}$ pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a “low” level to the $\overline{MCLR}$ pin, rather than pulling this pin directly to V_{SS}.

3: PORTD and PORTE are not implemented on the 28-pin devices.

† NOTICE: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

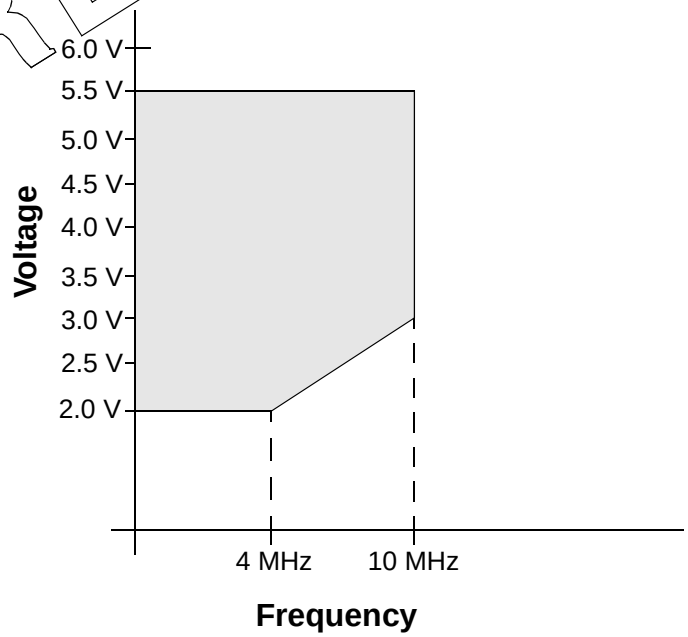
PRELIMINARY

FIGURE 15-1: PIC16FXXX-20 VOLTAGE-FREQUENCY GRAPH



PRELIMINARY

FIGURE 15-2: PIC16LFXXX-04 VOLTAGE-FREQUENCY GRAPH



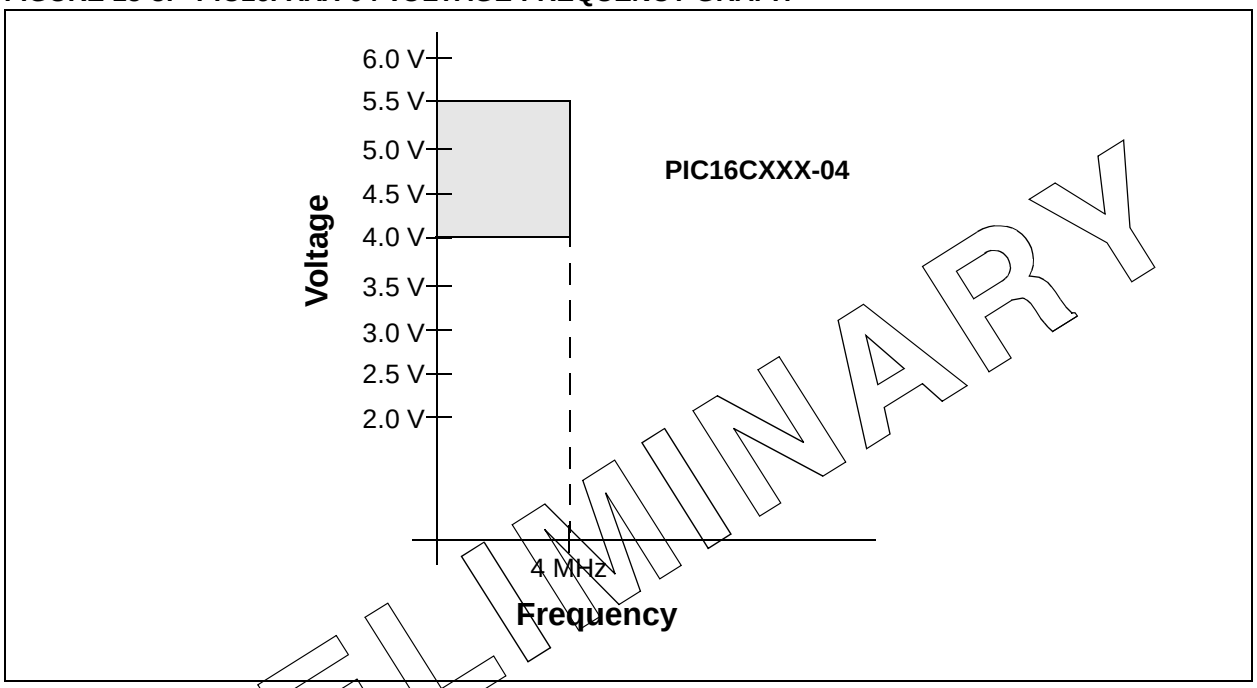
FMAX = (6.0 MHz/V) (VDDAPPMIN - 2.0 V) + 4 MHz

Note 1: VDDAPPMIN is the minimum voltage of the PICmicro® device in the application.

2: FMAX has a maximum frequency of 10MHz.

© 1999 Microchip Technology Inc.

FIGURE 15-3: PIC16FXXX-04 VOLTAGE-FREQUENCY GRAPH



PRELIMINARY

15.1 DC Characteristics: PIC16F873/874/876/877-04 (Commercial, Industrial) PIC16F873/874/876/877-20 (Commercial, Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial				
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001	Supply Voltage	VDD	4.0	-	5.5	V	XT, RC and LP osc configuration
D001A			4.5	-	5.5	V	HS osc configuration
			VBOR*	-	5.5	V	BOR enabled, Fmax = 14MHz (Note 7)
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	
D003	VDD start voltage to ensure internal Power-on Reset signal	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure internal Power-on Reset signal	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D005	Brown-out Reset Voltage	VBOR	3.7	4.0	4.35	V	BODEN bit in configuration word enabled
D010	Supply Current (Note 2,5)	IDD	-	1.6	4	mA	XT, RC osc configuration FOSC = 4 MHz, VDD = 5.5V (Note 4)
D013			-	7	15	mA	HS osc configuration FOSC = 20 MHz, VDD = 5.5V
D015*	Brown-out Reset Current (Note 6)	ΔIBOR	-	85	200	µA	BOR enabled VDD = 5.0V
D020	Power-down Current (Note 3,5)	IPD	-	10.5	42	µA	VDD = 4.0V, WDT enabled, -40°C to +85°C
D021			-	1.5	16	µA	VDD = 4.0V, WDT disabled, -0°C to +70°C
D021A			-	1.5	19	µA	VDD = 4.0V, WDT disabled, -40°C to +85°C
D023*	Brown-out Reset Current (Note 6)	ΔIBOR	-	85	200	µA	BOR enabled VDD = 5.0V

Legend: * These parameters are characterized but not tested.
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: This is the limit to which VDD can be lowered without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature also have an impact on the current consumption.
The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD
$\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula Ir = VDD/2Rext (mA) with Rext in kOhm.

5: Timer1 oscillator (when enabled) adds approximately 20 µA to the specification. This value is from characterization and is for design guidance only. This is not tested.

6: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

7: When BOR is enabled, the device will operate correctly until the VBOR voltage trip point is reached.

PRELIMINARY

© 1999 Microchip Technology Inc.

15.2 DC Characteristics: PIC16LF873/874/876/877-04 (Commercial, Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial				
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001	Supply Voltage	VDD	2.0	-	5.5	V	LP, XT, RC osc configuration (DC - 4 MHz)
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	
D003	VDD start voltage to ensure internal Power-on Reset signal	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure internal Power-on Reset signal	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D005	Brown-out Reset Voltage	VBOR	3.7	4.0	4.35	V	BODEN bit in configuration word enabled
D010	Supply Current (Note 2,5)	IDD	-	0.6	2.0	mA	XT, RC osc configuration FOSC = 4 MHz, VDD = 3.0V (Note 4)
D010A			-	20	35	µA	LP osc configuration FOSC = 32 kHz, VDD = 3.0V, WDT disabled
D015*	Brown-out Reset Current (Note 6)	ΔIBOR	-	85	200	µA	BOR enabled VDD = 5.0V
D020	Power-down Current (Note 3,5)	IPD	-	7.5	30	µA	VDD = 3.0V, WDT enabled, -40°C to +85°C
D021			-	0.9	5	µA	VDD = 3.0V, WDT disabled, 0°C to +70°C
D021A			-	0.9	5	µA	VDD = 3.0V, WDT disabled, -40°C to +85°C
D023*	Brown-out Reset Current (Note 6)	ΔIBOR	-	85	200	µA	BOR enabled VDD = 5.0V

Legend: * These parameters are characterized but not tested.
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: This is the limit to which VDD can be lowered without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature also have an impact on the current consumption.
The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD
MCLR = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula Ir = VDD/2Rext (mA) with Rext in kOhm.

5: Timer1 oscillator (when enabled) adds approximately 20 µA to the specification. This value is from characterization and is for design guidance only. This is not tested.

6: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

PRELIMINARY

15.3 DC Characteristics: PIC16F873/874/876/877-04 (Commercial, Industrial) PIC16F873/874/876/877-20 (Commercial, Industrial) PIC16LF873/874/876/877-04 (Commercial, Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial Operating voltage VDD range as described in DC spec Section 15.1 and Section 15.2.				
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
	Input Low Voltage						
	I/O ports	VIL					
D030	with TTL buffer		VSS	-	0.15VDD	V	For entire VDD range
D030A			VSS	-	0.8V	V	4.5V ≤ VDD ≤ 5.5V
D031	with Schmitt Trigger buffer		VSS	-	0.2VDD	V	
D032	MCLR, OSC1 (in RC mode)		VSS	-	0.2VDD	V	
D033	OSC1 (in XT, HS and LP)		VSS	-	0.3VDD	V	Note1
	Ports RC3 and RC4						
D034	with Schmitt Trigger buffer		VSS	-	0.3VDD	V	For entire VDD range
D034A	with SMBus		-0.5	-	0.6	V	for VDD = 4.5 to 5.5V
	Input High Voltage						
	I/O ports	VIH		-			
D040	with TTL buffer		2.0	-	VDD	V	4.5V ≤ VDD ≤ 5.5V
D040A			0.25VDD + 0.8V	-	VDD	V	For entire VDD range
D041	with Schmitt Trigger buffer		0.8VDD	-	VDD	V	For entire VDD range
D042	MCLR		0.8VDD	-	VDD	V	
D042A	OSC1 (XT, HS and LP)		0.7VDD	-	VDD	V	Note1
D043	OSC1 (in RC mode)		0.9VDD	-	VDD	V	
	Ports RC3 and RC4						
D044	with Schmitt Trigger buffer		0.7VDD	-	VDD	V	For entire VDD range
D044A	with SMBus		1.4	-	5.5	V	for VDD = 4.5 to 5.5V
D070	PORTB weak pull-up current	IPURB	50	250	400	µA	VDD = 5V, VPIN = VSS
	Input Leakage Current (Notes 2, 3)						
D060	I/O ports	IIL	-	-	±1	µA	Vss ≤ VPIN ≤ VDD, Pin at hi-impedance
D061	MCLR, RA4/T0CKI		-	-	±5	µA	Vss ≤ VPIN ≤ VDD
D063	OSC1		-	-	±5	µA	Vss ≤ VPIN ≤ VDD, XT, HS and LP osc configuration
	Output Low Voltage						
D080	I/O ports	VOL	-	-	0.6	V	IOL = 8.5 mA, VDD = 4.5V, -40°C to +85°C
D083	OSC2/CLKOUT (RC osc config)		-	-	0.6	V	IOL = 1.6 mA, VDD = 4.5V, -40°C to +85°C

Legend: * These parameters are characterized but not tested.
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16F87X be driven with external clock in RC mode.

2: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

PRELIMINARY

© 1999 Microchip Technology Inc.

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial Operating voltage VDD range as described in DC spec Section 15.1 and Section 15.2.				
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
	Output High Voltage						
D090	I/O ports (Note 3)	VOH	VDD - 0.7	-	-	V	IOH = -3.0 mA, VDD = 4.5V, -40°C to +85°C
D092	OSC2/CLKOUT (RC osc config)		VDD - 0.7	-	-	V	IOH = -1.3 mA, VDD = 4.5V, -40°C to +85°C
D150*	**Open-Drain High Voltage**	VOD	-	-	8.5	V	RA4 pin
			-	-			
	Capacitive Loading Specs on Output Pins						
D100	OSC2 pin	COSC2	-	-	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1.
D101	All I/O pins and OSC2 (in RC	CIO	-	-	50	pF	
D102	mode) SCL, SDA in I^2C mode	CB	-	-	400	pF	
	Data EEPROM Memory						
D120	Endurance	ED	100K	-	-	E/W	25°C at 5V
D121	VDD for read/write	VDRW	Vmin	-	5.5	V	Using EECON to read/write Vmin = min operating voltage
D122	Erase/write cycle time	TDEW	-	4	8	ms	
	Program FLASH Memory						
D130	Endurance	EP	1000	-	-	E/W	25°C at 5V
D131	VDD for read	VPR	Vmin	-	5.5	V	Vmin = min operating voltage
D132a	VDD for erase/write		Vmin	-	5.5	V	using EECON to read/write, Vmin = min operating voltage
D133	Erase/Write cycle time	TPEW	-	4	8	ms	

Legend: * These parameters are characterized but not tested.
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16F87X be driven with external clock in RC mode.

2: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

PRELIMINARY

15.4 Timing Parameter Symbology

The timing parameter symbols have been created following one of the following formats:

1. TppS2ppS
2. TppS
3. TCC:ST (I²C specifications only)
4. Ts (I²C specifications only)

T			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

pp			
cc	CCP1	osc	OSC1
ck	CLKOUT	rd	$\overline{RD}$
cs	$\overline{CS}$	rw	$\overline{RD}$ or $\overline{WR}$
di	SDI	sc	$\overline{SCK}$
do	SDO	ss	$\overline{SS}$
dt	Data in	t0	T0CKI
io	I/O port	t1	T1CKI
mc	$\overline{MCLR}$	wr	$\overline{WR}$

Uppercase letters and their meanings:

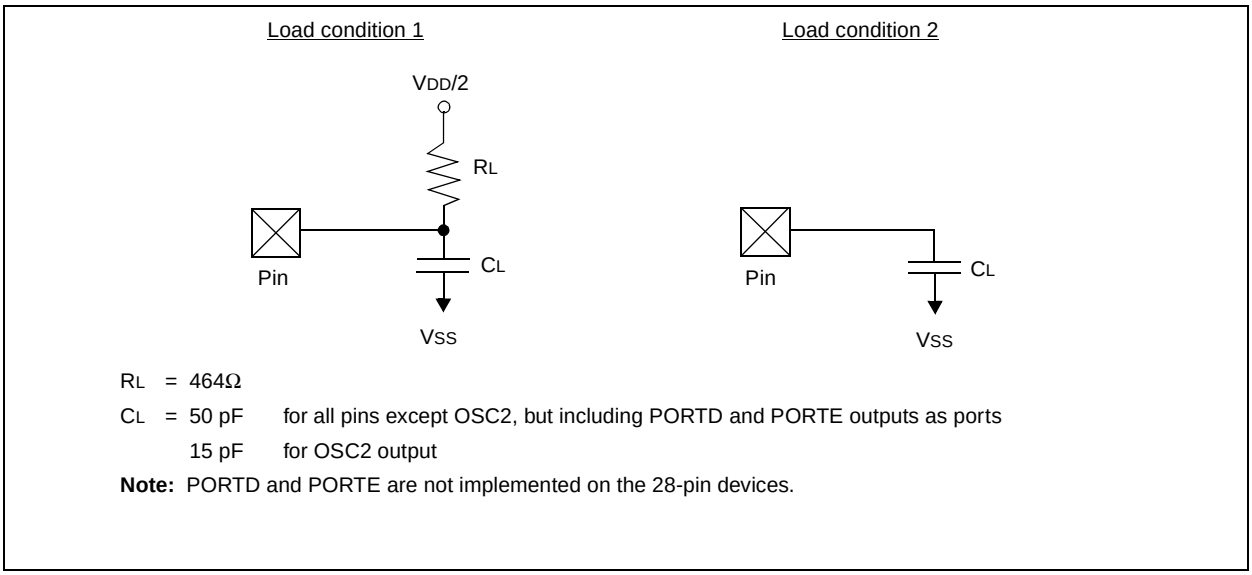
S			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
I²C only			
AA	output access	High	High
BUF	Bus free	Low	Low

TCC:ST (I²C specifications only)

CC			
HD	Hold	SU	Setup
ST			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

PRELIMINARY

FIGURE 15-4: LOAD CONDITIONS



RL = 464Ω

CL = 50 pF for all pins except OSC2, but including PORTD and PORTE outputs as ports

15 pF for OSC2 output

Note: PORTD and PORTE are not implemented on the 28-pin devices.

© 1999 Microchip Technology Inc.

FIGURE 15-5: EXTERNAL CLOCK TIMING

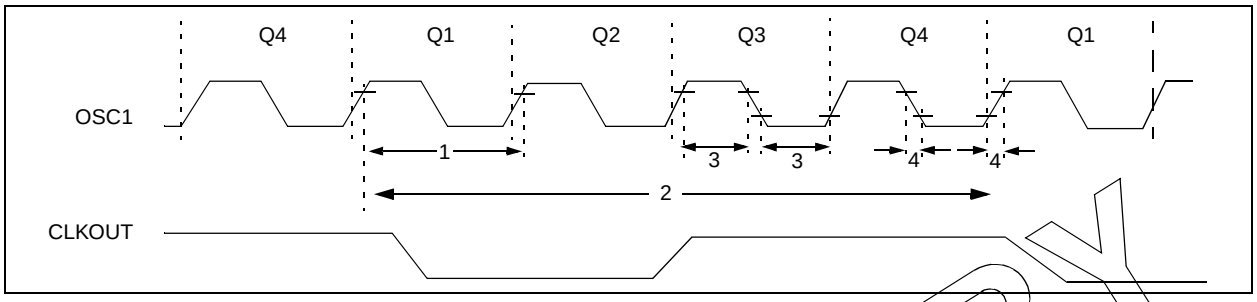


TABLE 15-1: EXTERNAL CLOCK TIMING REQUIREMENTS

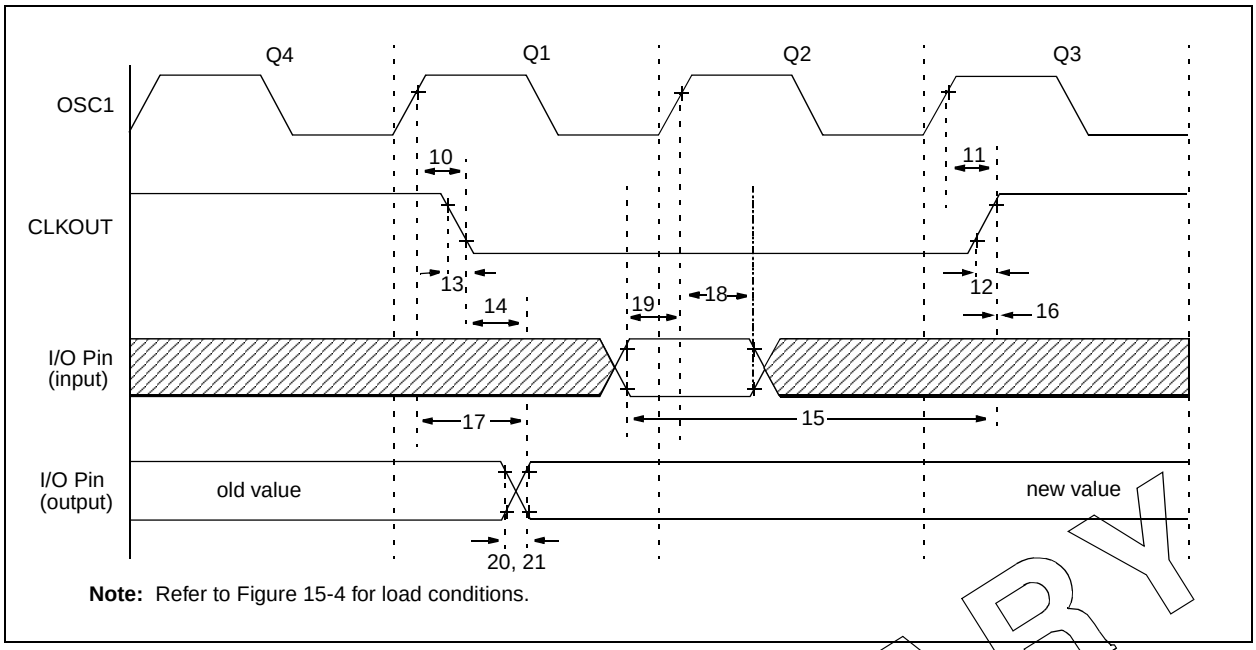
Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	FOSC	**External CLKIN Frequency (Note 1)**	DC	—	4	MHz	XT and RC osc mode
			DC	—	4	MHz	HS osc mode (-04)
			DC	—	20	MHz	HS osc mode (-20)
			DC	—	200	kHz	LP osc mode
		Oscillator Frequency (Note 1)	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			4	—	20	MHz	HS osc mode
			5	—	200	kHz	LP osc mode
1	TOSC	**External CLKIN Period (Note 1)**	250	—	—	ns	XT and RC osc mode
			250	—	—	ns	HS osc mode (-04)
			50	—	—	ns	HS osc mode (-20)
			5	—	—	µs	LP osc mode
		Oscillator Period (Note 1)	250	—	—	ns	RC osc mode
			250	—	10,000	ns	XT osc mode
			250	—	250	ns	HS osc mode (-04)
			50	—	250	ns	HS osc mode (-20)
			5	—	—	µs	LP osc mode
2	TCY	**Instruction Cycle Time (Note 1)**	200	TCY	DC	ns	TCY = 4/FOSC
3	TosL, TosH	**External Clock in (OSC1) High or Low Time**	100	—	—	ns	XT oscillator
			2.5	—	—	µs	LP oscillator
			15	—	—	ns	HS oscillator
4	TosR, TosF	**External Clock in (OSC1) Rise or Fall Time**	—	—	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator

Legend: † Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time-base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min." values with an external clock applied to the OSC1/CLKIN pin. When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices.

PRELIMINARY

FIGURE 15-6: CLKOUT AND I/O TIMING



Note: Refer to Figure 15-4 for load conditions.

TABLE 15-2: CLKOUT AND I/O TIMING REQUIREMENTS

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
10*	TosH2ckL	OSC1↑ to CLKOUT↓		—	75	200	ns	Note 1
11*	TosH2ckH	OSC1↑ to CLKOUT↑		—	75	200	ns	Note 1
12*	TckR	CLKOUT rise time		—	35	100	ns	Note 1
13*	TckF	CLKOUT fall time		—	35	100	ns	Note 1
14*	TckL2ioV	CLKOUT ↓ to Port out valid		—	—	0.5TCY + 20	ns	Note 1
15*	TioV2ckH	Port in valid before CLKOUT ↑		TOSC + 200	—	—	ns	Note 1
16*	TckH2ioI	Port in hold after CLKOUT ↑		0	—	—	ns	Note 1
17*	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid		—	100	255	ns	
18*	TosH2ioI	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	Standard (**F**)	100	—	—	ns	
			Extended (**LF**)	200	—	—	ns	
19*	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)		0	—	—	ns	
20*	TioR	Port output rise time	Standard (**F**)	—	10	40	ns	
			Extended (**LF**)	—	—	145	ns	
21*	TioF	Port output fall time	Standard (**F**)	—	10	40	ns	
			Extended (**LF**)	—	—	145	ns	
22††*	Tinp	INT pin high or low time		TCY	—	—	ns	
23††*	Trbp	RB7:RB4 change INT high or low time		TCY	—	—	ns	

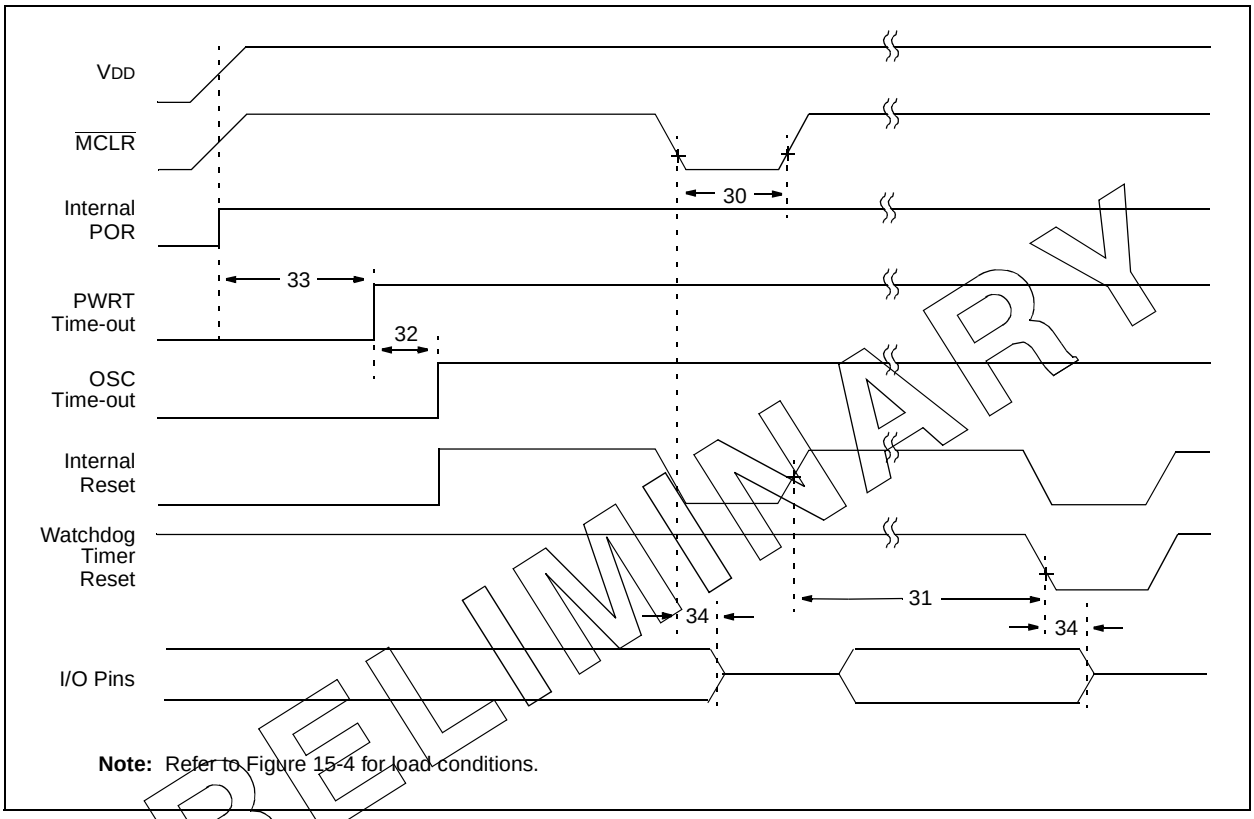
Legend: * These parameters are characterized but not tested.
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.
†† These parameters are asynchronous events not related to any internal clock edges.

Note 1: Measurements are taken in RC Mode where CLKOUT output is 4 x TOSC.

PRELIMINARY

© 1999 Microchip Technology Inc.

FIGURE 15-7: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING



Note: Refer to Figure 15-4 for load conditions.

PRELIMINARY

FIGURE 15-8: BROWN-OUT RESET TIMING

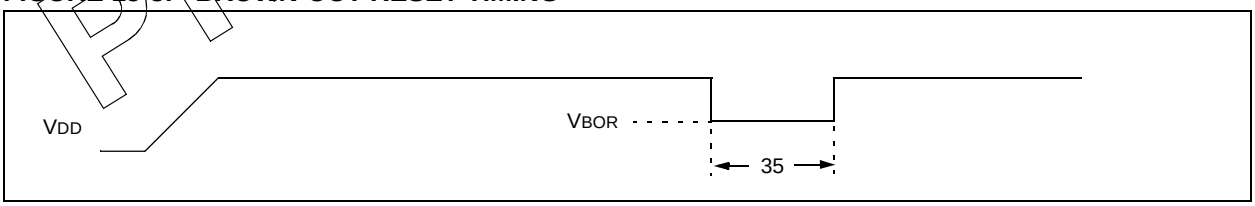
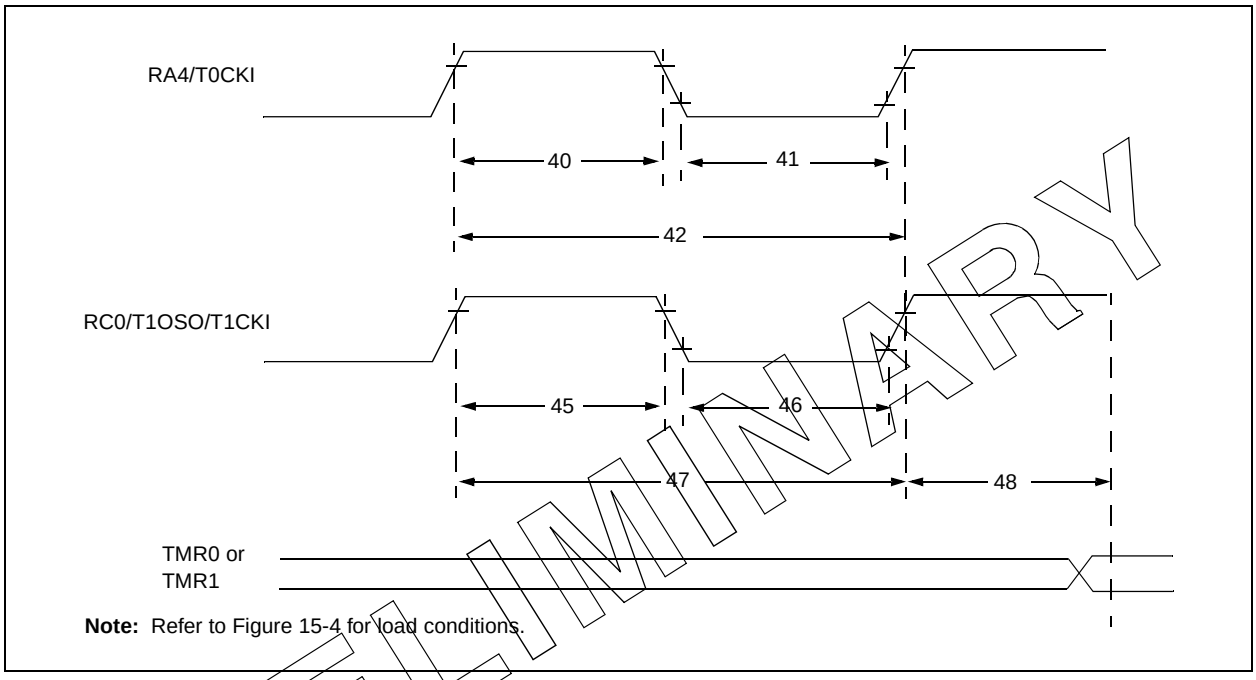


TABLE 15-3: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	TmcL	$\overline{MCLR}$ Pulse Width (low)	2	—	—	µs	VDD = 5V, -40°C to +85°C
31*	Twdt	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	VDD = 5V, -40°C to +85°C
32	Tost	Oscillation Start-up Timer Period	—	1024 TOSC	—	—	TOSC = OSC1 period
33*	Tpwrt	Power up Timer Period	28	72	132	ms	VDD = 5V, -40°C to +85°C
34	TIOZ	I/O Hi-impedance from $\overline{MCLR}$ Low or Watchdog Timer Reset	—	—	2.1	µs	
35	TBOR	Brown-out Reset pulse width	100	—	—	µs	VDD ≤ VBOR (D005)

Legend: * These parameters are characterized but not tested.
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 15-9: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS



Note: Refer to Figure 15-4 for load conditions.

PRELIMINARY

TABLE 15-4: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

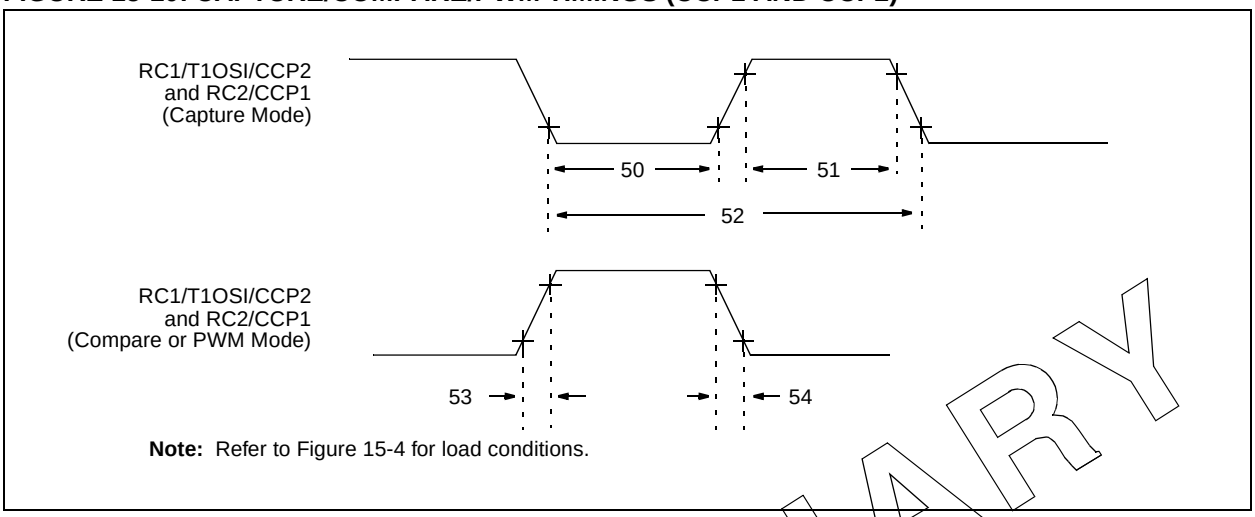
Param No.	Sym	Characteristic			Min	Typ†	Max	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width		No Prescaler	0.5TCY + 20	—	—	ns	Must also meet parameter 42
				With Prescaler	10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width		No Prescaler	0.5TCY + 20	—	—	ns	Must also meet parameter 42
				With Prescaler	10	—	—	ns	
42*	Tt0P	T0CKI Period		No Prescaler	TCY + 40	—	—	ns	
				With Prescaler	Greater of: 20 or TCY + 40 / N	—	—	ns	N = prescale value (2, 4, ..., 256)
45*	Tt1H	T1CKI High Time	Synchronous, Prescaler = 1		0.5TCY + 20	—	—	ns	Must also meet parameter 47
			Synchronous, Prescaler = 2,4,8	Standard(**F**)	15	—	—	ns	
				Extended(**LF**)	25	—	—	ns	
			Asynchronous	Standard(**F**)	30	—	—	ns	
				Extended(**LF**)	50	—	—	ns	
46*	Tt1L	T1CKI Low Time	Synchronous, Prescaler = 1		0.5TCY + 20	—	—	ns	Must also meet parameter 47
			Synchronous, Prescaler = 2,4,8	Standard(**F**)	15	—	—	ns	
				Extended(**LF**)	25	—	—	ns	
			Asynchronous	Standard(**F**)	30	—	—	ns	
				Extended(**LF**)	50	—	—	ns	
47*	Tt1P	T1CKI input period	Synchronous	Standard(**F**)	Greater of: 30 OR TCY + 40 / N	—	—	ns	N = prescale value (1, 2, 4, 8)
				Extended(**LF**)	Greater of: 50 OR TCY + 40 / N				N = prescale value (1, 2, 4, 8)
			Asynchronous	Standard(**F**)	60	—	—	ns	
				Extended(**LF**)	100	—	—	ns	
	Ft1	Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)			DC	—	200	kHz	
48	TCKEZtmr1	Delay from external clock edge to timer increment			2Tosc	—	7Tosc	—	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

© 1999 Microchip Technology Inc.

FIGURE 15-10: CAPTURE/COMPARE/PWM TIMINGS (CCP1 AND CCP2)



Note: Refer to Figure 15-4 for load conditions.

TABLE 15-5: CAPTURE/COMPARE/PWM REQUIREMENTS (CCP1 AND CCP2)

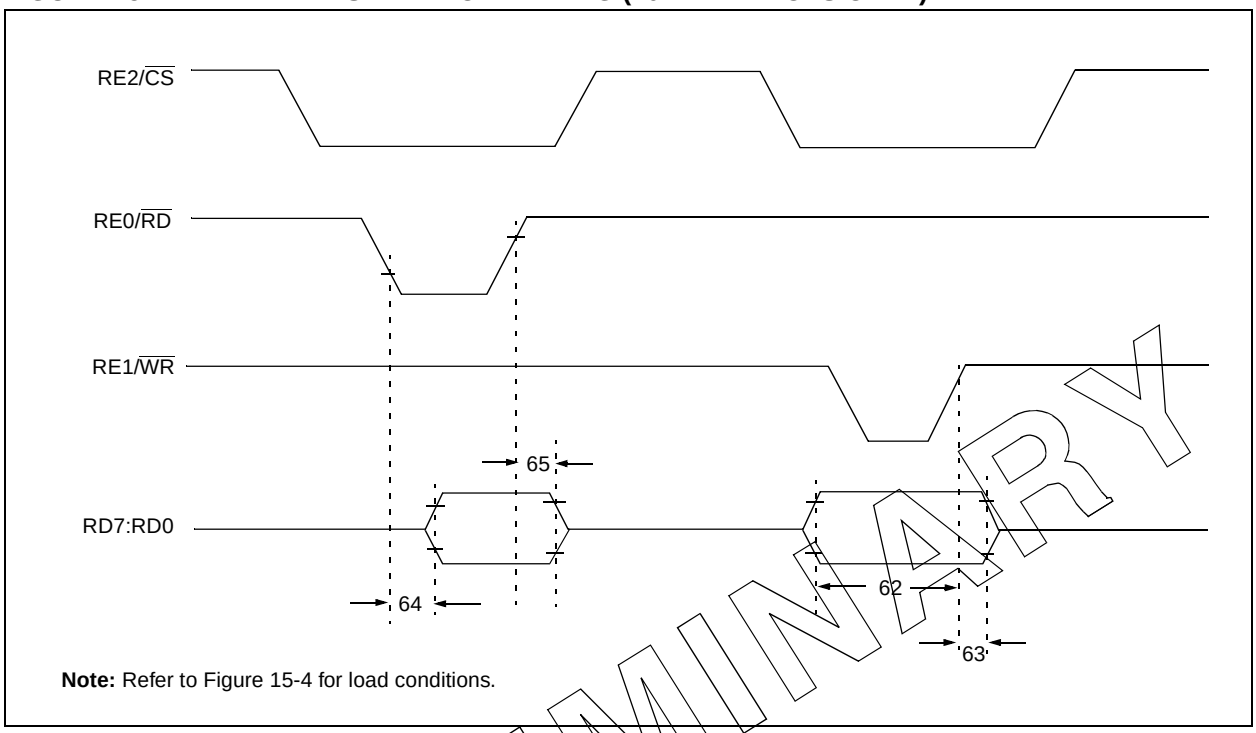
Param No.	Sym	Characteristic			Min	Typ†	Max	Units	Conditions
50*	TccL	CCP1 and CCP2 input low time	No Prescaler		0.5TCY + 20	—	—	ns	
			With Prescaler	Standard(**F**)	10	—	—	ns	
				Extended(**LF**)	20	—	—	ns	
51*	TccH	CCP1 and CCP2 input high time	No Prescaler		0.5TCY + 20	—	—	ns	
			With Prescaler	Standard(**F**)	10	—	—	ns	
				Extended(**LF**)	20	—	—	ns	
52*	TccP	CCP1 and CCP2 input period			$\frac{3T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1,4 or 16)
53*	TccR	CCP1 and CCP2 output rise time		Standard(**F**)	—	10	25	ns	
				Extended(**LF**)	—	25	50	ns	
54*	TccF	CCP1 and CCP2 output fall time		Standard(**F**)	—	10	25	ns	
				Extended(**LF**)	—	25	45	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated.
These parameters are for design guidance only and are not tested.

PRELIMINARY

FIGURE 15-11: PARALLEL SLAVE PORT TIMING (40-PIN DEVICES ONLY)



Note: Refer to Figure 15-4 for load conditions.

PRELIMINARY

TABLE 15-6: PARALLEL SLAVE PORT REQUIREMENTS (40-PIN DEVICES ONLY)

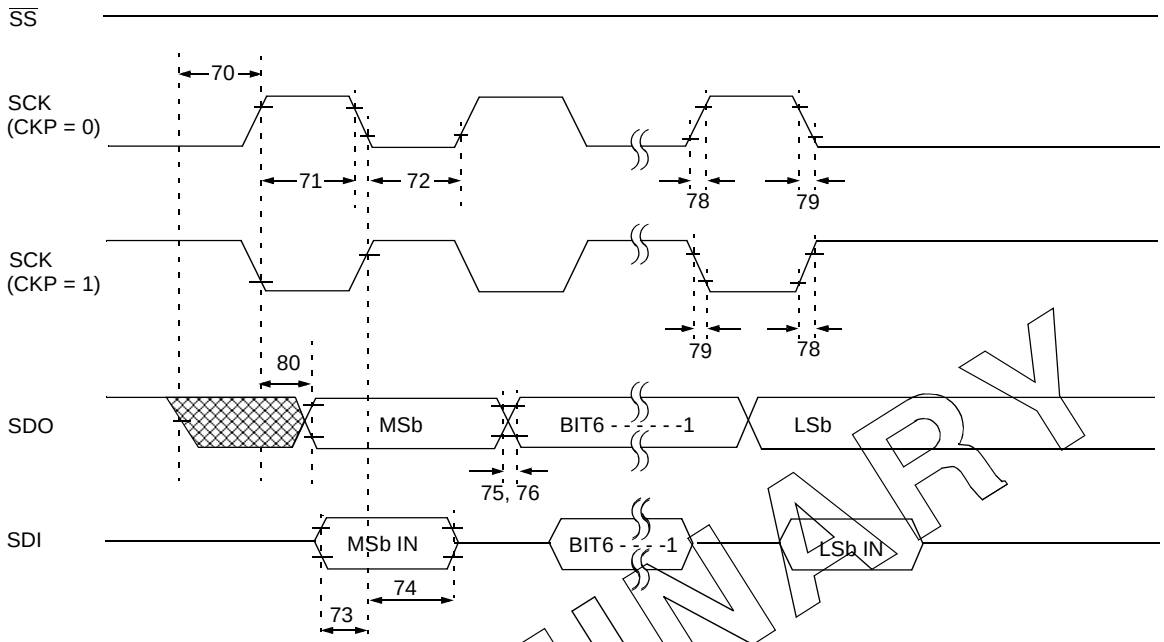
Parameter No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
62	TdtV2wrH	Data in valid before WR↑ or CS↑ (setup time)		20 25	— —	— —	ns ns	 Extended Range Only
63*	TwrH2dtI	WR↑ or CS↑ to data–in invalid (hold time)	Standard(**F**)	20	—	—	ns	
			Extended(**LF**)	35	—	—	ns	
64	TrdL2dtV	RD↓ and CS↓ to data–out valid		— —	— —	80 90	ns ns	 Extended Range Only
65	TrdH2dtI	RD↑ or CS↓ to data–out invalid		10	—	30	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

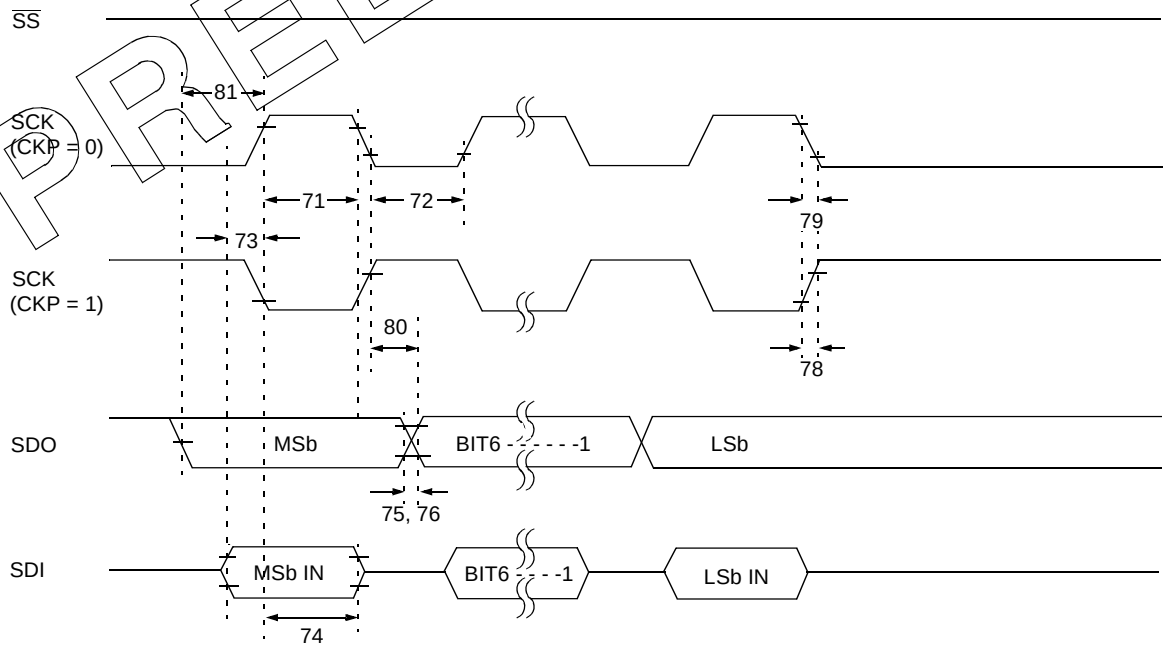
© 1999 Microchip Technology Inc.

FIGURE 15-12: SPI MASTER MODE TIMING (CKE = 0, SMP = 0)



Note: Refer to Figure 15-4 for load conditions.

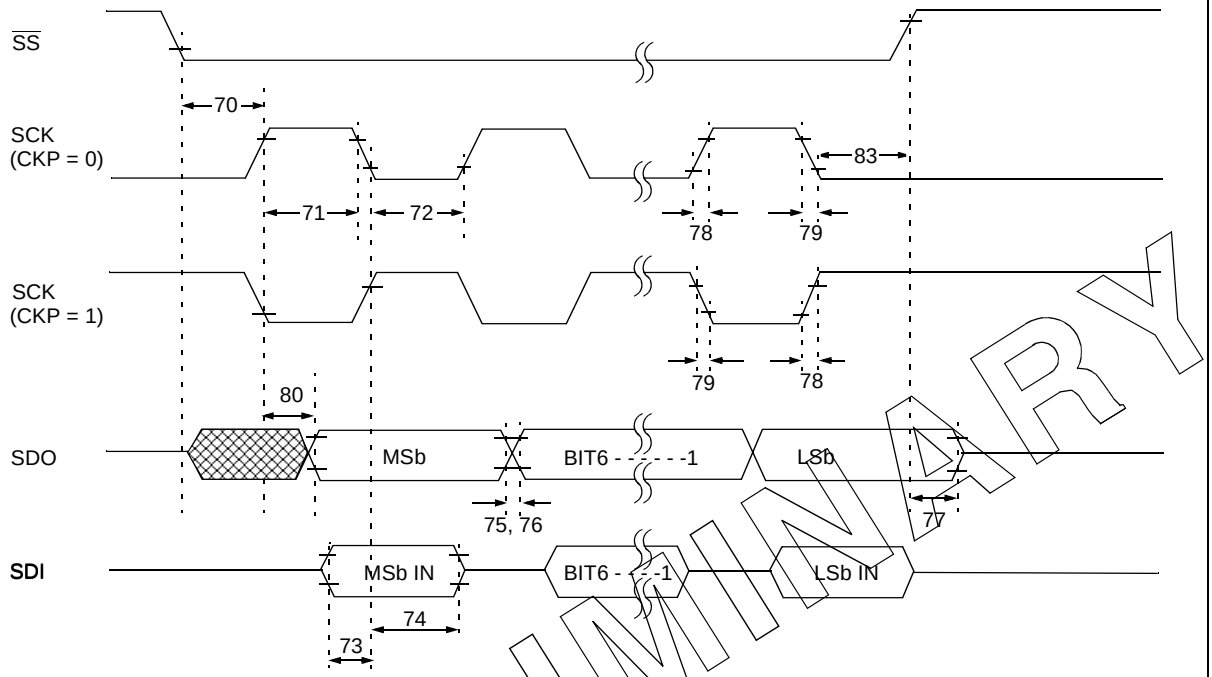
FIGURE 15-13: SPI MASTER MODE TIMING (CKE = 1, SMP = 1)



Note: Refer to Figure 15-4 for load conditions.

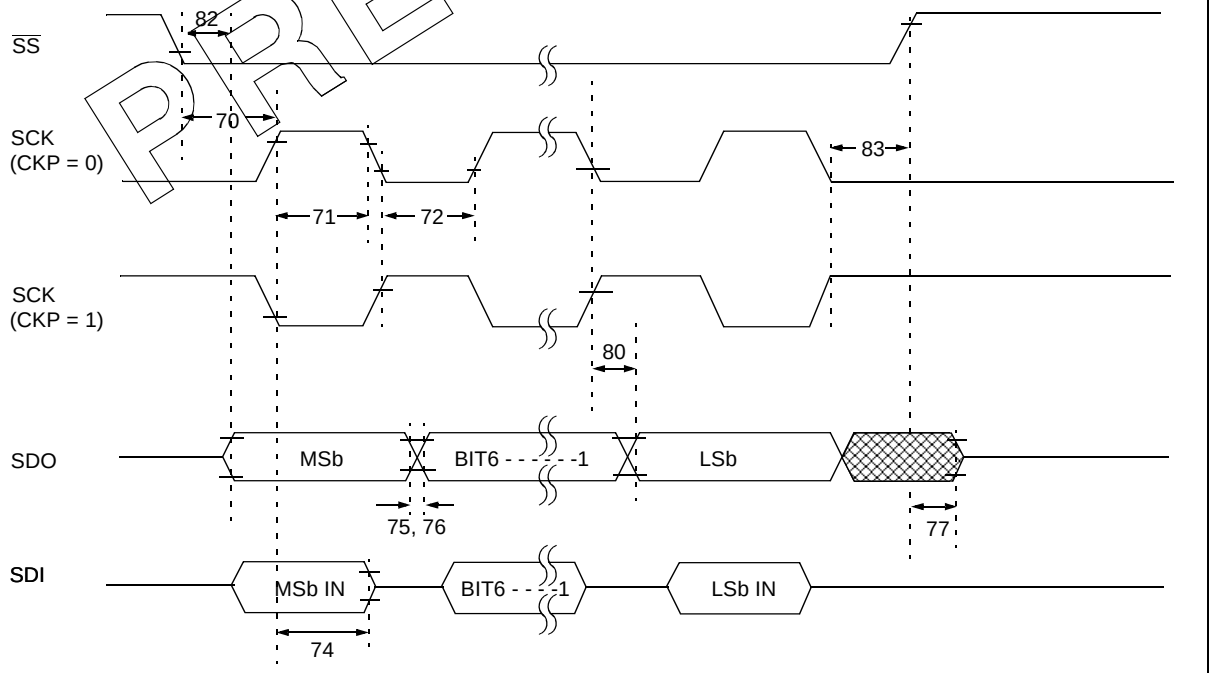
PRELIMINARY

FIGURE 15-14: SPI SLAVE MODE TIMING (CKE = 0)



Note: Refer to Figure 15-4 for load conditions.

FIGURE 15-15: SPI SLAVE MODE TIMING (CKE = 1)



Note: Refer to Figure 15-4 for load conditions.

PRELIMINARY

© 1999 Microchip Technology Inc.

TABLE 15-7: SPI MODE REQUIREMENTS

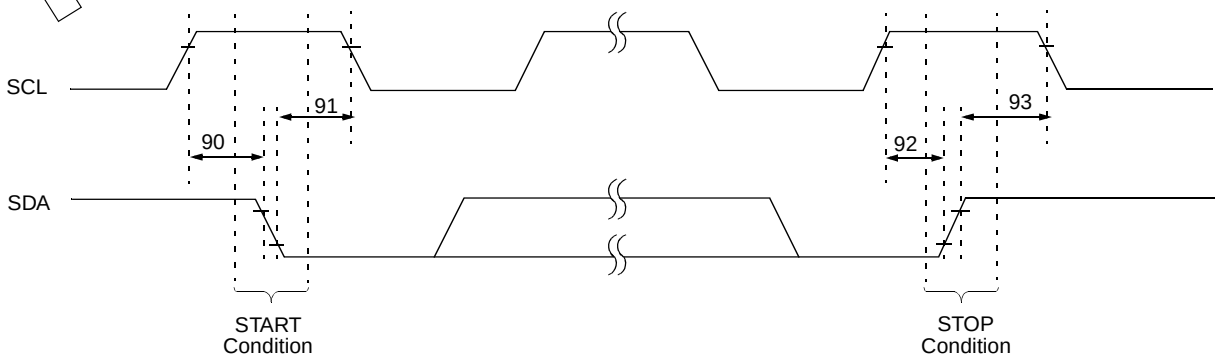
Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
70*	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ to SCK$\downarrow$ or SCK$\uparrow$ input		TCY	—	—	ns	
71*	TscH	SCK input high time (slave mode)		TCY + 20	—	—	ns	
72*	TscL	SCK input low time (slave mode)		TCY + 20	—	—	ns	
73*	TdiV2scH, TdiV2scL	Setup time of SDI data input to SCK edge		100	—	—	ns	
74*	TscH2diL, TscL2diL	Hold time of SDI data input to SCK edge		100	—	—	ns	
75*	TdoR	SDO data output rise time	Standard(**F**) Extended(**LF**)	— —	10 25	25 50	ns ns	
76*	TdoF	SDO data output fall time		—	10	25	ns	
77*	TssH2doZ	$\overline{SS}\uparrow$ to SDO output hi-impedance		10	—	50	ns	
78*	TscR	SCK output rise time (master mode)	Standard(**F**) Extended(**LF**)	— —	10 25	25 50	ns ns	
79*	TscF	SCK output fall time (master mode)		—	10	25	ns	
80*	TscH2doV, TscL2doV	SDO data output valid after SCK edge	Standard(**F**) Extended(**LF**)	— —	— —	50 145	ns	
81*	TdoV2scH, TdoV2scL	SDO data output setup to SCK edge		TCY	—	—	ns	
82*	TssL2doV	SDO data output valid after $\overline{SS}\downarrow$ edge		—	—	50	ns	
83*	TscH2ssH, TscL2ssH	$\overline{SS}\uparrow$ after SCK edge		1.5TCY + 40	—	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

PRELIMINARY

FIGURE 15-16: I^2C BUS START/STOP BITS TIMING

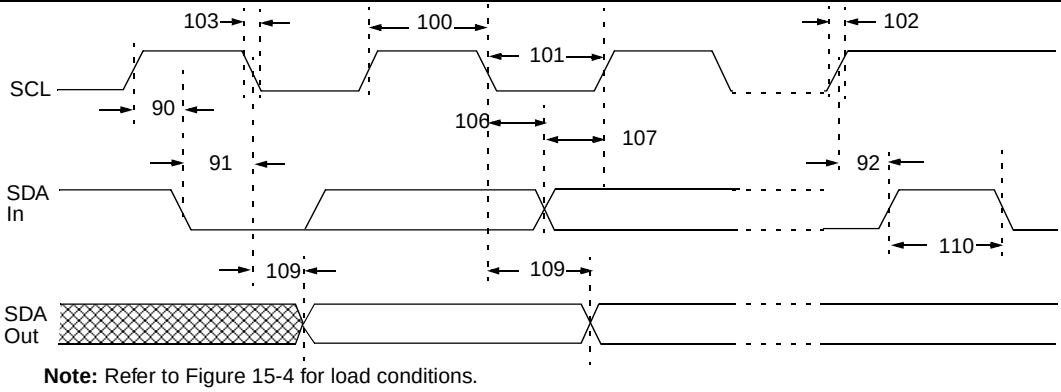


Note: Refer to Figure 15-4 for load conditions.

TABLE 15-8: I^2C BUS START/STOP BITS REQUIREMENTS

Parameter No.	Sym	Characteristic		Min	Typ	Max	Units	Conditions
90	TSU:STA	START condition Setup time	100 kHz mode	4700	—	—	ns	Only relevant for repeated START condition
			400 kHz mode	600	—	—		
91	THD:STA	START condition Hold time	100 kHz mode	4000	—	—	ns	After this period the first clock pulse is generated
			400 kHz mode	600	—	—		
92	TSU:STO	STOP condition Setup time	100 kHz mode	4700	—	—	ns	
			400 kHz mode	600	—	—		
93	THD:STO	STOP condition Hold time	100 kHz mode	4000	—	—	ns	
			400 kHz mode	600	—	—		

FIGURE 15-17: I^2C BUS DATA TIMING



Note: Refer to Figure 15-4 for load conditions.

TABLE 15-9: I^2C BUS DATA REQUIREMENTS

Param No.	Sym	Characteristic		Min	Max	Units	Conditions
100	THIGH	Clock high time	100 kHz mode	4.0	—	µs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	µs	Device must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
101	TLOW	Clock low time	100 kHz mode	4.7	—	µs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	µs	Device must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
102	TR	SDA and SCL rise time	100 kHz mode	—	1000	ns	
			400 kHz mode	20 + 0.1Cb	300	ns	Cb is specified to be from 10 to 400 pF
103	TF	SDA and SCL fall time	100 kHz mode	—	300	ns	
			400 kHz mode	20 + 0.1Cb	300	ns	Cb is specified to be from 10 to 400 pF
90	TSU:STA	START condition setup time	100 kHz mode	4.7	—	µs	Only relevant for repeated START condition
			400 kHz mode	0.6	—	µs	
91	THD:STA	START condition hold time	100 kHz mode	4.0	—	µs	After this period the first clock pulse is generated
			400 kHz mode	0.6	—	µs	
106	THD:DAT	Data input hold time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	µs	
107	TSU:DAT	Data input setup time	100 kHz mode	250	—	ns	Note 2
			400 kHz mode	100	—	ns	
92	TSU:STO	STOP condition setup time	100 kHz mode	4.7	—	µs	
			400 kHz mode	0.6	—	µs	
109	TAA	Output valid from clock	100 kHz mode	—	3500	ns	Note 1
			400 kHz mode	—	—	ns	
110	TBUF	Bus free time	100 kHz mode	4.7	—	µs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	µs	
	Cb	Bus capacitive loading		—	400	pF	

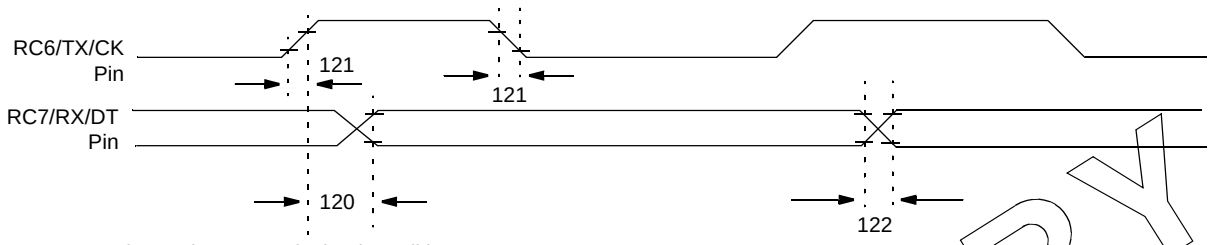
Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.

2: A fast-mode (400 kHz) I^2C-bus device can be used in a standard-mode (100 kHz) I^2C-bus system, but the requirement tsu; DAT ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line TR max.+tsu; DAT = 1000 + 250 = 1250 ns (according to the standard-mode I^2C bus specification) before the SCL line is released.

PRELIMINARY

© 1999 Microchip Technology Inc.

FIGURE 15-18: USART SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING



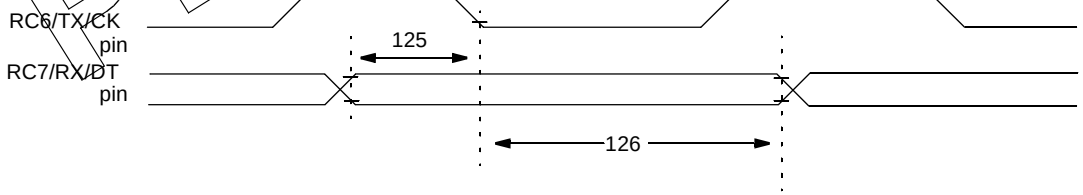
Note: Refer to Figure 15-4 for load conditions.

TABLE 15-10: USART SYNCHRONOUS TRANSMISSION REQUIREMENTS

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
120	TckH2dtV	SYNC XMIT (MASTER & SLAVE) Clock high to data out valid	Standard(**F**)	—	—	80	ns	
			Extended(**LF**)	—	—	100	ns	
121	Tckrf	Clock out rise time and fall time (Master Mode)	Standard(**F**)	—	—	45	ns	
			Extended(**LF**)	—	—	50	ns	
122	Tdtrf	Data out rise time and fall time	Standard(**F**)	—	—	45	ns	
			Extended(**LF**)	—	—	50	ns	

†: Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 15-19: USART SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING



Note: Refer to Figure 15-4 for load conditions.

TABLE 15-11: USART SYNCHRONOUS RECEIVE REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
125	TdtV2ckL	SYNC RCV (MASTER & SLAVE) Data setup before CK ↓ (DT setup time)	15	—	—	ns	
126	TckL2dtl	Data hold after CK ↓ (DT hold time)	15	—	—	ns	

†: Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

PRELIMINARY

**TABLE 15-12: PIC16F873/874/876/877-04 (COMMERCIAL, INDUSTRIAL)
PIC16F873/874/876/877-20 (COMMERCIAL, INDUSTRIAL)
PIC16LF873/874/876/877-04 (COMMERCIAL, INDUSTRIAL)**

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
A01	NR	Resolution		—	—	10-bits	bit	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A03	EIL	Integral linearity error		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A04	EDL	Differential linearity error		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A06	EOFF	Offset error		—	—	< ± 2	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A07	EGN	Gain error		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A10	—	Monotonicity(3)		—	guaranteed	—	—	VSS ≤ VAIN ≤ VREF
A20	VREF	Reference voltage (VREF+ - VREF-)		2.0V	—	VDD + 0.3	V	Absolute minimum electrical spec. To ensure 10-bit accuracy.
A21	VREF+	Reference voltage High		AVDD - 2.5V		AVDD + 0.3V	V	
A22	VREF-	Reference voltage low		AVSS - 0.3V		VREF+ - 2.0V	V	
A25	VAIN	Analog input voltage		VSS - 0.3	—	VREF + 0.3	V	
A30	ZAIN	Recommended impedance of analog voltage source		—	—	10.0	kΩ	
A40	IAD	A/D conversion current (VDD)	Standard	—	220	—	µA	Average current consumption when A/D is on. (Note 1)
			Extended	—	90	—	µA	
A50	IREF	VREF input current (Note 2)		10	—	1000	µA	During VAIN acquisition. Based on differential of VHOLD to VAIN to charge CHOLD, see Section 11.1.
				—	—	10	µA	During A/D Conversion cycle

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: When A/D is off, it will not consume any current other than minor leakage current. The power-down current spec includes any such leakage from the A/D module.

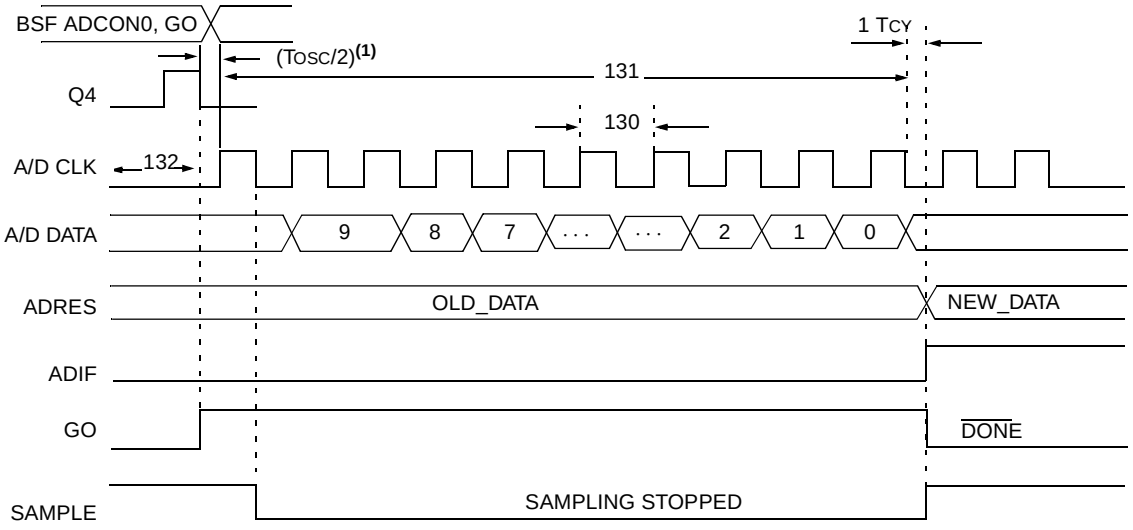
2: VREF current is from RA3 pin or VDD pin, whichever is selected as reference input.

3: The A/D conversion result never decreases with an increase in the Input Voltage, and has no missing codes.

PRELIMINARY

© 1999 Microchip Technology Inc.

FIGURE 15-20: A/D CONVERSION TIMING



Note 1: If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

TABLE 15-13: A/D CONVERSION REQUIREMENTS

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
130	TAD	A/D clock period	Standard(**F**)	1.6	—	—	µs	TOSC based, VREF ≥ 3.0V
			Extended(**LF**)	3.0	—	—	µs	TOSC based, VREF ≥ 2.0V
			Standard(**F**)	2.0	4.0	6.0	µs	A/D RC Mode
			Extended(**LF**)	3.0	6.0	9.0	µs	A/D RC Mode
131	TCNV	Conversion time (not including S/H time) (Note 1)			—	12	TAD	
132	TACQ	Acquisition time		Note 2	40	—	µs	
				10*	—	—	µs	The minimum time is the amplifier settling time. This may be used if the "new" input voltage has not changed by more than 1 LSb (i.e., 20.0 mV @ 5.12V) from the last sampled voltage (as stated on CHOLD).
134	TGO	Q4 to A/D clock start		—	TOSC/2 §	—	—	If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

§ This specification ensured by design.

Note 1: ADRES register may be read on the following TCY cycle.

2: See Section 11.1 for min conditions.

NOTES:

© 1999 Microchip Technology Inc.

16.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES

The graphs and tables provided in this section are for **design guidance** and are **not tested**.

In some graphs or tables, the data presented are **outside specified operating range** (i.e., outside specified V_{DD} range). This is for **information only** and devices are ensured to operate properly only within the specified range.

The data presented in this section is a **statistical summary** of data collected on units from different lots over a period of time and matrix samples. 'Typical' represents the mean of the distribution at 25°C. 'Max' or 'min' represents (mean + 3σ) or (mean - 3σ) respectively, where σ is standard deviation, over the whole temperature range.

Graphs and Tables not available at this time.

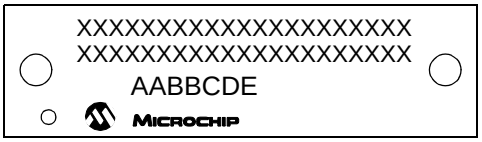
NOTES:

© 1999 Microchip Technology Inc.

17.0 PACKAGING INFORMATION

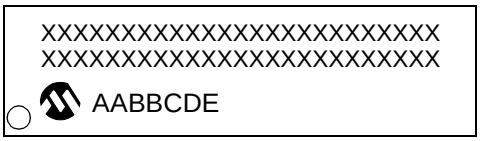
17.1 Package Marking Information

28-Lead PDIP (Skinny DIP)

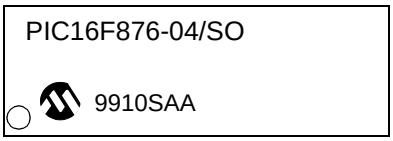


Example

28-Lead SOIC



Example



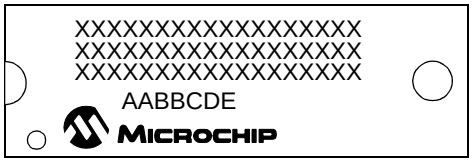
Legend:	MM...M	Microchip part number information
	XX...X	Customer specific information*
	AA	Year code (last 2 digits of calendar year)
	BB	Week code (week of January 1 is week '01')
	C	Facility code of the plant at which wafer is manufactured O = Outside Vendor C = 5" Line S = 6" Line H = 8" Line
	D	Mask revision number
	E	Assembly code of the plant or country of origin in which part was assembled

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line thus limiting the number of available characters for customer specific information.

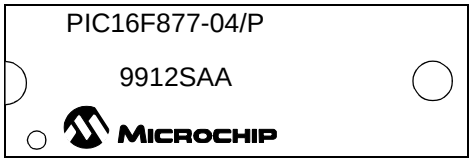
***** Standard OTP marking consists of Microchip part number, year code, week code, facility code, mask rev#, and assembly code. For OTP marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

Package Marking Information (Cont'd)

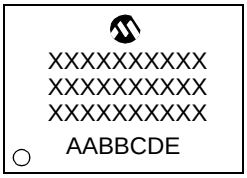
40-Lead PDIP



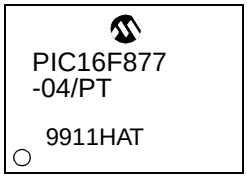
Example



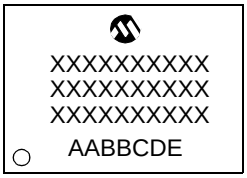
44-Lead TQFP



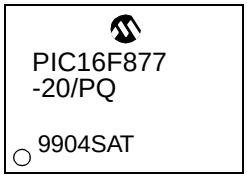
Example



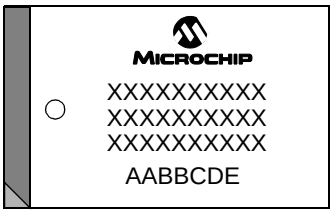
44-Lead MQFP



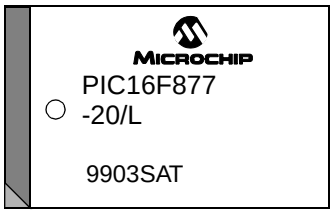
Example



44-Lead PLCC

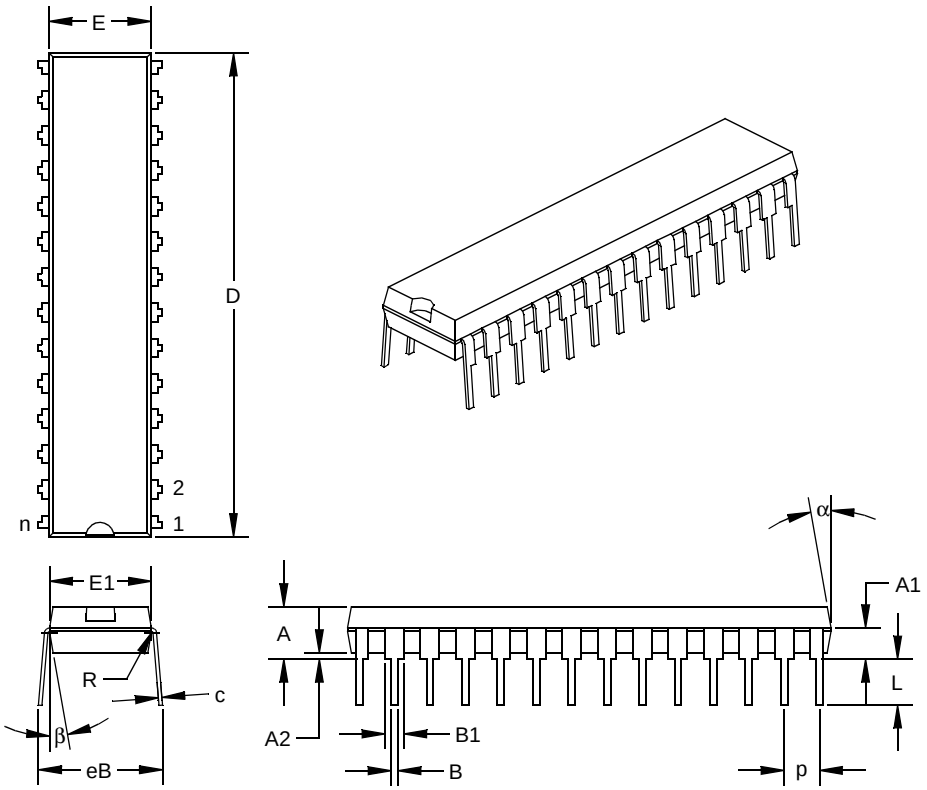


Example



© 1999 Microchip Technology Inc.

17.2 K04-070 28-Lead Skinny Plastic Dual In-line (SP) – 300 mil



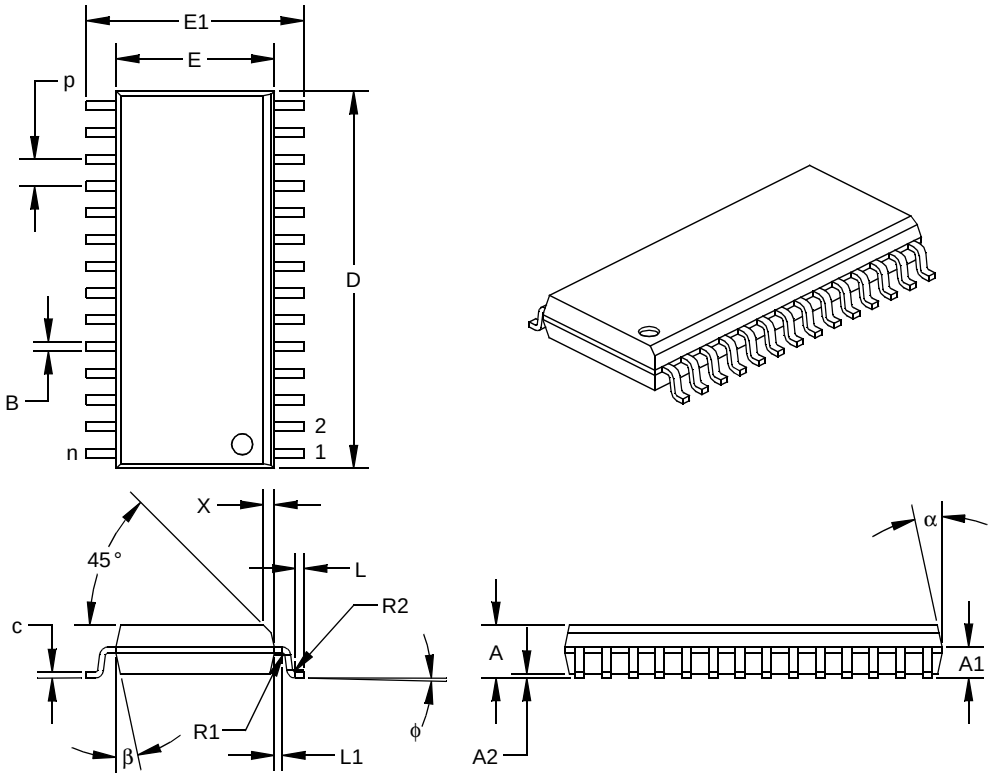
Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
PCB Row Spacing			0.300			7.62	
Number of Pins	n		28			28	
Pitch	p		0.100			2.54	
Lower Lead Width	B	0.016	0.019	0.022	0.41	0.48	0.56
Upper Lead Width	B1†	0.040	0.053	0.065	1.02	1.33	1.65
Shoulder Radius	R	0.000	0.005	0.010	0.00	0.13	0.25
Lead Thickness	c	0.008	0.010	0.012	0.20	0.25	0.30
Top to Seating Plane	A	0.140	0.150	0.160	3.56	3.81	4.06
Top of Lead to Seating Plane	A1	0.070	0.090	0.110	1.78	2.29	2.79
Base to Seating Plane	A2	0.015	0.020	0.025	0.38	0.51	0.64
Tip to Seating Plane	L	0.125	0.130	0.135	3.18	3.30	3.43
Package Length	D‡	1.345	1.365	1.385	34.16	34.67	35.18
Molded Package Width	E‡	0.280	0.288	0.295	7.11	7.30	7.49
Radius to Radius Width	E1	0.270	0.283	0.295	6.86	7.18	7.49
Overall Row Spacing	eB	0.320	0.350	0.380	8.13	8.89	9.65
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

* Controlling Parameter.

† Dimension “B1” does not include dam-bar protrusions. Dam-bar protrusions shall not exceed 0.003” (0.076 mm) per side or 0.006” (0.152 mm) more than dimension “B1.”

‡ Dimensions “D” and “E” do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.010” (0.254 mm) per side or 0.020” (0.508 mm) more than dimensions “D” or “E.”

17.3 K04-052 28-Lead Plastic Small Outline (SO) – Wide, 300 mil



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Pitch	p		0.050			1.27	
Number of Pins	n		28			28	
Overall Pack. Height	A	0.093	0.099	0.104	2.36	2.50	2.64
Shoulder Height	A1	0.048	0.058	0.068	1.22	1.47	1.73
Standoff	A2	0.004	0.008	0.011	0.10	0.19	0.28
Molded Package Length	D‡	0.700	0.706	0.712	17.78	17.93	18.08
Molded Package Width	E‡	0.292	0.296	0.299	7.42	7.51	7.59
Outside Dimension	E1	0.394	0.407	0.419	10.01	10.33	10.64
Chamfer Distance	X	0.010	0.020	0.029	0.25	0.50	0.74
Shoulder Radius	R1	0.005	0.005	0.010	0.13	0.13	0.25
Gull Wing Radius	R2	0.005	0.005	0.010	0.13	0.13	0.25
Foot Length	L	0.011	0.016	0.021	0.28	0.41	0.53
Foot Angle	ϕ	0	4	8	0	4	8
Radius Centerline	L1	0.010	0.015	0.020	0.25	0.38	0.51
Lead Thickness	c	0.009	0.011	0.012	0.23	0.27	0.30
Lower Lead Width	B†	0.014	0.017	0.019	0.36	0.42	0.48
Mold Draft Angle Top	α	0	12	15	0	12	15
Mold Draft Angle Bottom	β	0	12	15	0	12	15

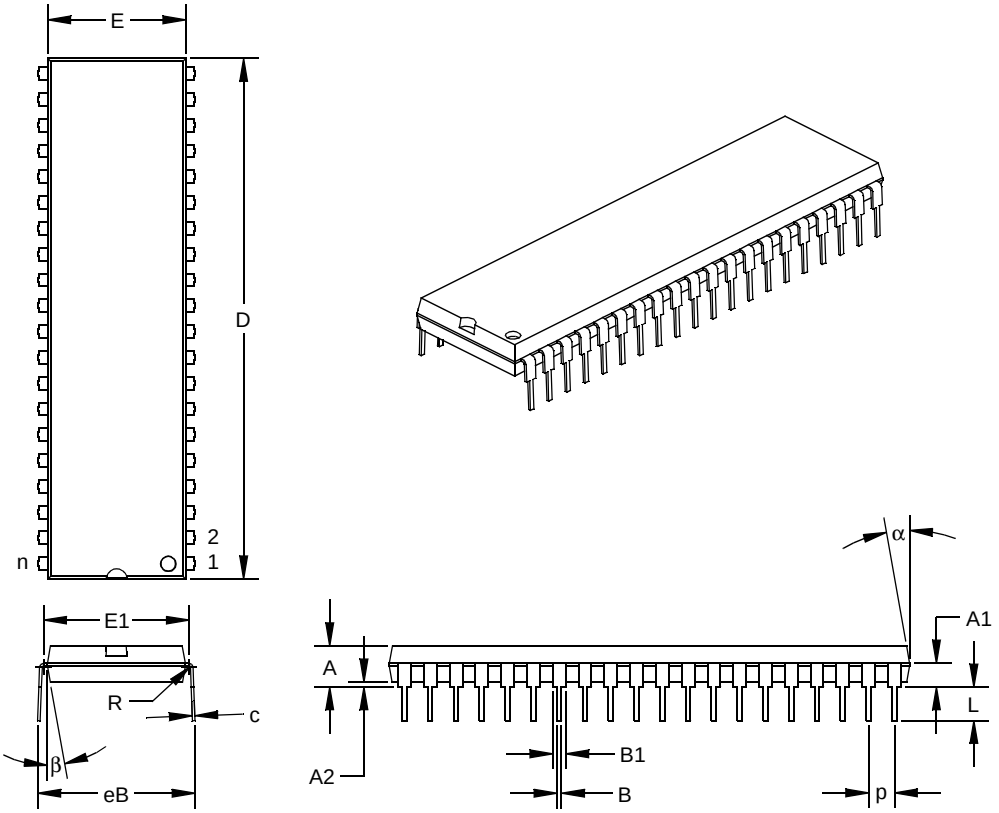
* Controlling Parameter.

† Dimension “B” does not include dam-bar protrusions. Dam-bar protrusions shall not exceed 0.003” (0.076 mm) per side or 0.006” (0.152 mm) more than dimension “B.”

‡ Dimensions “D” and “E” do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.010” (0.254 mm) per side or 0.020” (0.508 mm) more than dimensions “D” or “E.”

© 1999 Microchip Technology Inc.

17.4 K04-016 40-Lead Plastic Dual In-line (P) – 600 mil



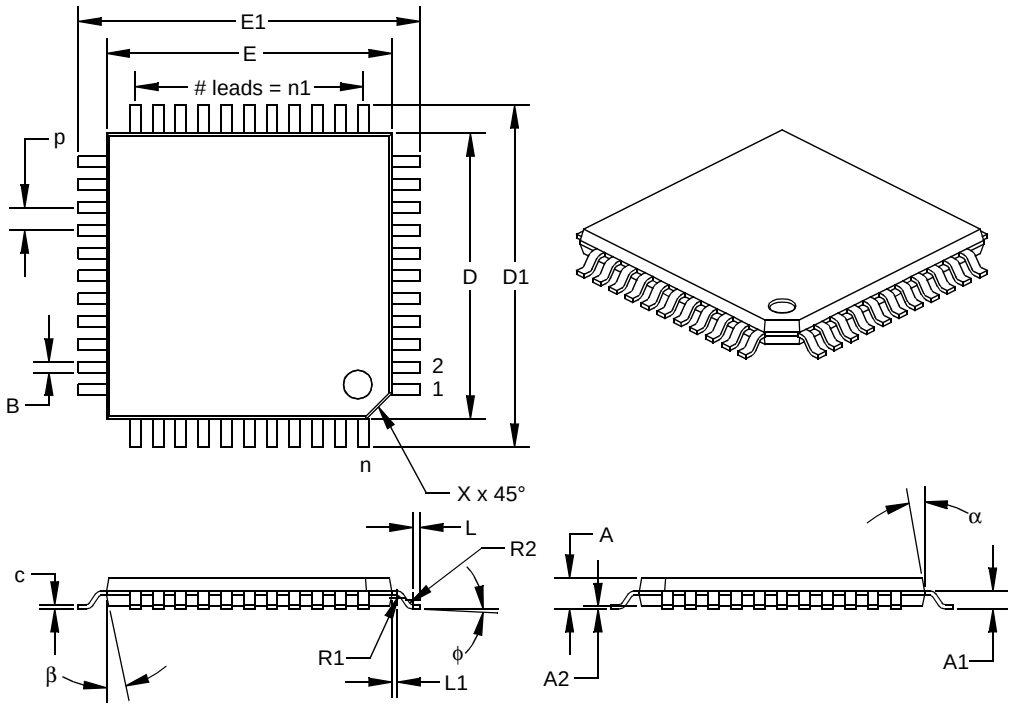
Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
PCB Row Spacing			0.600			15.24	
Number of Pins	n		40			40	
Pitch	p		0.100			2.54	
Lower Lead Width	B	0.016	0.018	0.020	0.41	0.46	0.51
Upper Lead Width	B1†	0.045	0.050	0.055	1.14	1.27	1.40
Shoulder Radius	R	0.000	0.005	0.010	0.00	0.13	0.25
Lead Thickness	c	0.009	0.010	0.011	0.23	0.25	0.28
Top to Seating Plane	A	0.110	0.160	0.160	2.79	4.06	4.06
Top of Lead to Seating Plane	A1	0.073	0.093	0.113	1.85	2.36	2.87
Base to Seating Plane	A2	0.020	0.020	0.040	0.51	0.51	1.02
Tip to Seating Plane	L	0.125	0.130	0.135	3.18	3.30	3.43
Package Length	D‡	2.013	2.018	2.023	51.13	51.26	51.38
Molded Package Width	E‡	0.530	0.535	0.540	13.46	13.59	13.72
Radius to Radius Width	E1	0.545	0.565	0.585	13.84	14.35	14.86
Overall Row Spacing	eB	0.630	0.610	0.670	16.00	15.49	17.02
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

* Controlling Parameter.

† Dimension “B1” does not include dam-bar protrusions. Dam-bar protrusions shall not exceed 0.003” (0.076 mm) per side or 0.006” (0.152 mm) more than dimension “B1.”

‡ Dimensions “D” and “E” do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.010” (0.254 mm) per side or 0.020” (0.508 mm) more than dimensions “D” or “E.”

17.5 K04-076 44-Lead Plastic Thin Quad Flatpack (PT) 10x10x1 mm Body, 1.0/0.1 mm Lead Form



Units		INCHES			MILLIMETERS*		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Pitch	p		0.031			0.80	
Number of Pins	n		44			44	
Pins along Width	n1		11			11	
Overall Pack. Height	A	0.039	0.043	0.047	1.00	1.10	1.20
Shoulder Height	A1	0.015	0.025	0.035	0.38	0.64	0.89
Standoff	A2	0.002	0.004	0.006	0.05	0.10	0.15
Shoulder Radius	R1	0.003	0.003	0.010	0.08	0.08	0.25
Gull Wing Radius	R2	0.003	0.006	0.008	0.08	0.14	0.20
Foot Length	L	0.005	0.010	0.015	0.13	0.25	0.38
Foot Angle	ϕ	0	3.5	7	0	3.5	7
Radius Centerline	L1	0.003	0.008	0.013	0.08	0.20	0.33
Lead Thickness	c	0.004	0.006	0.008	0.09	0.15	0.20
Lower Lead Width	B†	0.012	0.015	0.018	0.30	0.38	0.45
Outside Tip Length	D1	0.463	0.472	0.482	11.75	12.00	12.25
Outside Tip Width	E1	0.463	0.472	0.482	11.75	12.00	12.25
Molded Pack. Length	D‡	0.390	0.394	0.398	9.90	10.00	10.10
Molded Pack. Width	E‡	0.390	0.394	0.398	9.90	10.00	10.10
Pin 1 Corner Chamfer	X	0.025	0.035	0.045	0.64	0.89	1.14
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	12	15	5	12	15

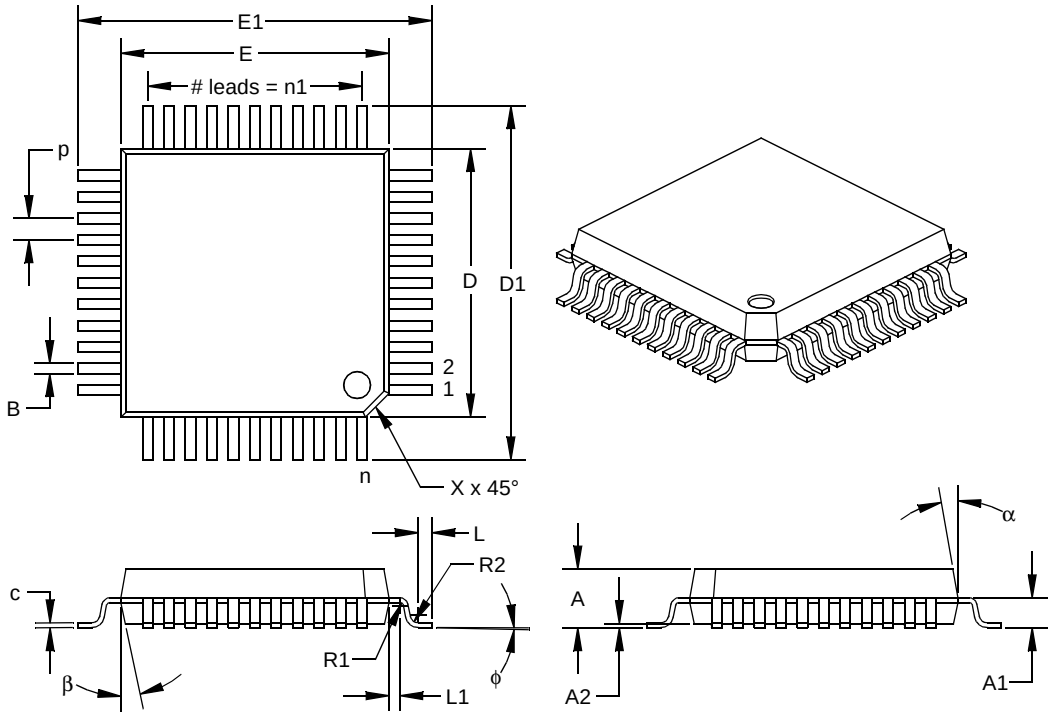
* Controlling Parameter.

† Dimension “B” does not include dam-bar protrusions. Dam-bar protrusions shall not exceed 0.003” (0.076 mm) per side or 0.006” (0.152 mm) more than dimension “B.”

‡ Dimensions “D” and “E” do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.010” (0.254 mm) per side or 0.020” (0.508 mm) more than dimensions “D” or “E.”
JEDEC equivalent:MS-026 ACB

© 1999 Microchip Technology Inc.

17.6 K04-071 44-Lead Plastic Quad Flatpack (PQ) 10x10x2 mm Body, 1.6/0.15 mm Lead Form



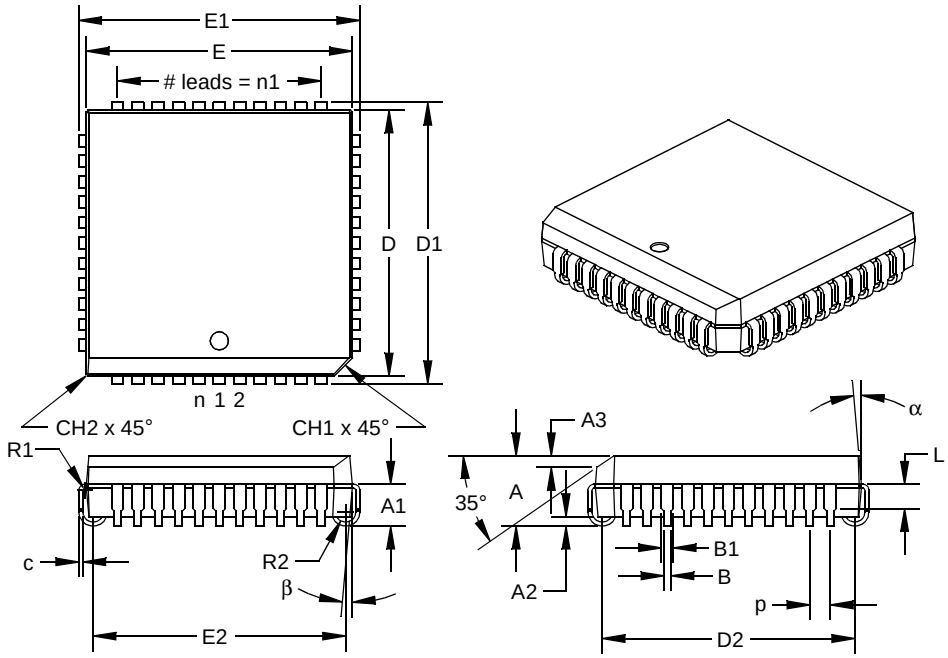
Units		INCHES			MILLIMETERS*		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Pitch	p		0.031			0.80	
Number of Pins	n		44			44	
Pins along Width	n1		11			11	
Overall Pack. Height	A	0.079	0.086	0.093	2.00	2.18	2.35
Shoulder Height	A1	0.032	0.044	0.056	0.81	1.11	1.41
Standoff	A2	0.002	0.006	0.010	0.05	0.15	0.25
Shoulder Radius	R1	0.005	0.005	0.010	0.13	0.13	0.25
Gull Wing Radius	R2	0.005	0.012	0.015	0.13	0.30	0.38
Foot Length	L	0.015	0.020	0.025	0.38	0.51	0.64
Foot Angle	ϕ	0	3.5	7	0	3.5	7
Radius Centerline	L1	0.011	0.016	0.021	0.28	0.41	0.53
Lead Thickness	c	0.005	0.007	0.009	0.13	0.18	0.23
Lower Lead Width	B†	0.012	0.015	0.018	0.30	0.37	0.45
Outside Tip Length	D1	0.510	0.520	0.530	12.95	13.20	13.45
Outside Tip Width	E1	0.510	0.520	0.530	12.95	13.20	13.45
Molded Pack. Length	D‡	0.390	0.394	0.398	9.90	10.00	10.10
Molded Pack. Width	E‡	0.390	0.394	0.398	9.90	10.00	10.10
Pin 1 Corner Chamfer	X	0.025	0.035	0.045	0.635	0.89	1.143
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	12	15	5	12	15

* Controlling Parameter.

† Dimension “B” does not include dam-bar protrusions. Dam-bar protrusions shall not exceed 0.003” (0.076 mm) per side or 0.006” (0.152 mm) more than dimension “B.”

‡ Dimensions “D” and “E” do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.010” (0.254 mm) per side or 0.020” (0.508 mm) more than dimensions “D” or “E.”
JEDEC equivalent:MS-022 AB

17.7 K04-048 44-Lead Plastic Leaded Chip Carrier (L) – Square



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		44			44	
Pitch	p		0.050			1.27	
Overall Pack. Height	A	0.165	0.173	0.180	4.19	4.38	4.57
Shoulder Height	A1	0.095	0.103	0.110	2.41	2.60	2.79
Standoff	A2	0.015	0.023	0.030	0.38	0.57	0.76
Side 1 Chamfer Dim.	A3	0.024	0.029	0.034	0.61	0.74	0.86
Corner Chamfer (1)	CH1	0.040	0.045	0.050	1.02	1.14	1.27
Corner Chamfer (other)	CH2	0.000	0.005	0.010	0.00	0.13	0.25
Overall Pack. Width	E1	0.685	0.690	0.695	17.40	17.53	17.65
Overall Pack. Length	D1	0.685	0.690	0.695	17.40	17.53	17.65
Molded Pack. Width	E‡	0.650	0.653	0.656	16.51	16.59	16.66
Molded Pack. Length	D‡	0.650	0.653	0.656	16.51	16.59	16.66
Footprint Width	E2	0.610	0.620	0.630	15.49	15.75	16.00
Footprint Length	D2	0.610	0.620	0.630	15.49	15.75	16.00
Pins along Width	n1		11			11	
Lead Thickness	c	0.008	0.010	0.012	0.20	0.25	0.30
Upper Lead Width	B1†	0.026	0.029	0.032	0.66	0.74	0.81
Lower Lead Width	B	0.015	0.018	0.021	0.38	0.46	0.53
Upper Lead Length	L	0.050	0.058	0.065	1.27	1.46	1.65
Shoulder Inside Radius	R1	0.003	0.005	0.010	0.08	0.13	0.25
J-Bend Inside Radius	R2	0.015	0.025	0.035	0.38	0.64	0.89
Mold Draft Angle Top	α	0	5	10	0	5	10
Mold Draft Angle Bottom	β	0	5	10	0	5	10

* Controlling Parameter.

† Dimension “B1” does not include dam-bar protrusions. Dam-bar protrusions shall not exceed 0.003” (0.076 mm) per side or 0.006” (0.152 mm) more than dimension “B1.”

‡ Dimensions “D” and “E” do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.010" (0.254 mm) per side or 0.020" (0.508 mm) more than dimensions “D” or “E.”
JEDEC equivalent:MO-047 AC

© 1999 Microchip Technology Inc.

APPENDIX A: REVISION HISTORY

Version	Date	Revision Description
A	1998	This is a new data sheet. However, these devices are similar to the PIC16C7X devices found in the PIC16C7X Data Sheet (DS30390). Data Memory Map for PIC16F873/874, moved ADFM bit from ADCON1<5> to ADCON1<7>
B	1999	FLASH EEPROM access information.

APPENDIX B: DEVICE DIFFERENCES

The differences between the devices in this data sheet are listed in Table B-1.

TABLE B-1: DEVICE DIFFERENCES

Difference	PIC16F876/873	PIC16F877/874
A/D	5 channels, 10bits	8 channels, 10bits
Parallel Slave Port	no	yes
Packages	28-pin PDIP, 28-pin windowed CERDIP, 28-pin SOIC	40-pin PDIP, 44-pin TQFP, 44-pin MQFP, 44-pin PLCC

APPENDIX C: CONVERSION CONSIDERATIONS

Considerations for converting from previous versions of devices to the ones listed in this data sheet are listed in Table C-1.

TABLE C-1: CONVERSION CONSIDERATIONS

Characteristic	PIC16C7X	PIC16F87X
Pins	28/40	28/40
Timers	3	3
Interrupts	11 or 12	13 or 14
Communication	PSP, USART, SSP (SPI, I^2C Slave)	PSP, USART, SSP (SPI, I^2C Master/Slave)
Frequency	20 MHz	20 MHz
A/D	8-bit	10-bit
CCP	2	2
Program Memory	4K, 8K EPROM	4K, 8K FLASH
RAM	192, 368 bytes	192, 368 bytes
EEPROM data	None	128, 256 bytes
Other	—	In-Circuit Debugger, Low Voltage Programming

© 1999 Microchip Technology Inc.

NOTES:

© 1999 Microchip Technology Inc.

INDEX

A

A/D 111
ADCON0 Register 111
ADCON1 Register 112
ADIF bit 113
Analog Input Model Block Diagram 115
Analog Port Pins 7, 8, 9, 37, 38
Block Diagram 114
Configuring Analog Port Pins 116
Configuring the Interrupt 113
Configuring the Module 113
Conversion Clock 116
Conversions 117
Delays 115
Effects of a Reset 118
GO/$\overline{\text{DONE}}$ bit 113
Internal Sampling Switch (Rss) Impedence 114
Operation During Sleep 118
Sampling Requirements 114
Source Impedence 114
Time Delays 115
Absolute Maximum Ratings 151
$\overline{\text{ACK}}$ 72
Acknowledge Data bit 66
Acknowledge Pulse 72
Acknowledge Sequence Enable bit 66
Acknowledge Status bit 66
ADRES Register 15, 111
Application Note AN578, "Use of the SSP Module in the I2C Multi-Master Environment." 71
Application Notes
AN552 (Implementing Wake-up on Key Strokes Using PIC16CXXX) 31
AN556 (Table Reading Using PIC16CXX) 26
Architecture
PIC16F873/PIC16F876 Block Diagram 5
PIC16F874/PIC16F877 Block Diagram 6
Assembler
MPASM Assembler 145

B

Banking, Data Memory 12, 18
Baud Rate Generator 78
BCLIF 24
BF 64, 72, 81, 83
Block Diagrams
A/D 114
Analog Input Model 115
Baud Rate Generator 78
Capture 59
Compare 60
I^2C Master Mode 76
I^2C Module 71
PWM 60
SSP (I^2C Mode) 71
SSP (SPI Mode) 67
Timer0/WDT Prescaler 47
Timer2 55
USART Receive 101
USART Transmit 99
BRG 78
BRGH bit 97
Brown-out Reset (BOR) 121, 125, 127, 128
BOR Status ($\overline{\text{BOR}}$ Bit) 25
Buffer Full bit, BF 72
Buffer Full Status bit, BF 64
Bus Arbitration 88
Bus Collision Section 88
Bus Collision During a RESTART Condition 91
Bus Collision During a Start Condition 89
Bus Collision During a Stop Condition 92
Bus Collision Interrupt Flag bit, BCLIF 24

C

Capture/Compare/PWM
Capture
Block Diagram 59
CCP1CON Register 58
CCP1IF 59
Mode 59
Prescaler 59
CCP Timer Resources 57
Compare
Block Diagram 60
Mode 60
Software Interrupt Mode 60
Special Event Trigger 60
Special Trigger Output of CCP1 60
Special Trigger Output of CCP2 60
Interaction of Two CCP Modules 57
Section 57
Special Event Trigger and A/D Conversions 60
Capture/Compare/PWM (CCP)
CCP1
RC2/CCP1 Pin 7, 8
CCP2
RC1/T1OSI/CCP2 Pin 7, 8
PWM Block Diagram 60
PWM Mode 60
CCP1CON 17
CCP2CON 17
CCPR1H Register 15, 17, 57
CCPR1L Register 17, 57
CCPR2H Register 15, 17
CCPR2L Register 15, 17
CCPxM0 bit 58
CCPxM1 bit 58
CCPxM2 bit 58
CCPxM3 bit 58
CCPxX bit 58
CCPxY bit 58
CKE 64
CKP 65
Clock Polarity Select bit, CKP 65
Code Examples
Call of a Subroutine in Page 1 from Page 0 26
Indirect Addressing 27
Code Protection 121, 135
Computed GOTO 26
Configuration Bits 121
Conversion Considerations 183

D

D/$\overline{\text{A}}$ 64
Data Memory 12
Bank Select (RP1:RP0 Bits) 12, 18
General Purpose Registers 12
Register File Map 13, 14
Special Function Registers 15
Data/$\overline{\text{Address}}$ bit, D/$\overline{\text{A}}$ 64
DC Characteristics 154

Development Support 145
Device Differences 183
Device Overview 5
Direct Addressing 27, 28

E

Electrical Characteristics 151
Errata 4

F

Firmware Instructions 137
FSR Register 15, 16, 17, 27

G

General Call Address Sequence 74
General Call Address Support 74
General Call Enable bit 66

I

I/O Ports 29
I^2C 71
I^2C Master Mode Reception 83
I^2C Master Mode Restart Condition 80
I^2C Mode Selection 71
I^2C Module
- Acknowledge Sequence timing 85
- Addressing 72
- Baud Rate Generator 78
- Block Diagram 76
- BRG Block Diagram 78
- BRG Reset due to SDA Collision 90
- BRG Timing 78
- Bus Arbitration 88
- Bus Collision 88
  - Acknowledge 88
  - Restart Condition 91
  - Restart Condition Timing (Case1) 91
  - Restart Condition Timing (Case2) 91
  - Start Condition 89
  - Start Condition Timing 89, 90
  - Stop Condition 92
  - Stop Condition Timing (Case1) 92
  - Stop Condition Timing (Case2) 92
  - Transmit Timing 88
- Bus Collision timing 88
- Clock Arbitration 87
- Clock Arbitration Timing (Master Transmit) 87
- Conditions to not give $\overline{ACK}$ Pulse 72
- General Call Address Support 74
- Master Mode 76
- Master Mode 7-bit Reception timing 84
- Master Mode Operation 77
- Master Mode Start Condition 79
- Master Mode Transmission 81
- Master Mode Transmit Sequence 77
- Multi-Master Communication 88
- Multi-master Mode 77
- Operation 71
- Repeat Start Condition timing 80
- Slave Mode 72
- Slave Reception 73
- Slave Transmission 73
- SSPBUF 72
- Stop Condition Receive or Transmit timing 86
- Stop Condition timing 86
- Waveforms for 7-bit Reception 73
- Waveforms for 7-bit Transmission 74

I^2C Module Address Register, SSPADD 72
I^2C Slave Mode 72
ID Locations 121, 135
In-Circuit Serial Programming (ICSP) 121, 136
INDF 17
INDF Register 15, 16, 27
Indirect Addressing 27, 28
- FSR Register 12

Instruction Format 137
Instruction Set 137
- ADDLW 139
- ADDWF 139
- ANDLW 139
- ANDWF 139
- BCF 139
- BSF 139
- BTFSC 140
- BTFSS 140
- CALL 140
- CLRF 140
- CLRW 140
- CLRWDT 140
- COMF 141
- DECF 141
- DECFSZ 141
- GOTO 141
- INCF 141
- INCFSZ 141
- IORLW 142
- IORWF 142
- MOVF 142
- MOVLW 142
- MOVWF 142
- NOP 142
- RETFIE 143
- RETLW 143
- RETURN 143
- RLF 143
- RRF 143
- SLEEP 143
- SUBLW 144
- SUBWF 144
- SWAPF 144
- XORLW 144
- XORWF 144
- Summary Table 138

INTCON 17
INTCON Register 20
- GIE Bit 20
- INTE Bit 20
- INTF Bit 20
- PEIE Bit 20
- RBIE Bit 20
- RBIF Bit 20, 31
- T0IE Bit 20
- T0IF Bit 20

Inter-Integrated Circuit (I^2C) 63
Internal Sampling Switch (Rss) Impedence 114
Interrupt Sources 121, 131
- Block Diagram 131
- Interrupt on Change (RB7:RB4) 31
- RB0/INT Pin, External 7, 8, 132
- TMR0 Overflow 132
- USART Receive/Transmit Complete 95

© 1999 Microchip Technology Inc.

Interrupts
Bus Collision Interrupt 24
Synchronous Serial Port Interrupt 22
Interrupts, Context Saving During 132
Interrupts, Enable Bits
Global Interrupt Enable (GIE Bit) 20, 131
Interrupt on Change (RB7:RB4) Enable (RBIE Bit) 20, 132
Peripheral Interrupt Enable (PEIE Bit) 20
RB0/INT Enable (INTE Bit) 20
TMR0 Overflow Enable (T0IE Bit) 20
Interrupts, Flag Bits
Interrupt on Change (RB7:RB4) Flag (RBIF Bit) 20, 31, 132
RB0/INT Flag (INTF Bit) 20
TMR0 Overflow Flag (T0IF Bit) 20, 132

K

KeeLoq® Evaluation and Programming Tools 148

L

Loading of PC 26

M

Master Clear ($\overline{MCLR}$) 7, 8
$\overline{MCLR}$ Reset, Normal Operation 125, 127, 128
$\overline{MCLR}$ Reset, SLEEP 125, 127, 128
Memory Organization
Data Memory 12
Program Memory 11
MPLAB Integrated Development Environment Software . 145
Multi-Master Communication 88
Multi-Master Mode 77

O

OPCODE Field Descriptions 137
OPTION 17
OPTION_REG Register 19
INTEDG Bit 19
PS2:PS0 Bits 19
PSA Bit 19
$\overline{RBPU}$ Bit 19
T0CS Bit 19
T0SE Bit 19
OSC1/CLKIN Pin 7, 8
OSC2/CLKOUT Pin 7, 8
Oscillator Configuration 121, 123
HS 123, 127
LP 123, 127
RC 123, 124, 127
XT 123, 127
Oscillator, WDT 133
Output of TMR2 55

P

P 64
Packaging 175
Paging, Program Memory 11, 26
Parallel Slave Port (PSP) 9, 35, 38
Block Diagram 38
RE0/$\overline{RD}$/AN5 Pin 9, 37, 38
RE1/$\overline{WR}$/AN6 Pin 9, 37, 38
RE2/$\overline{CS}$/AN7 Pin 9, 37, 38
Read Waveforms 39
Select (PSPMODE Bit) 35, 36, 38
Write Waveforms 39
PCL Register 15, 16, 17, 26
PCLATH Register 15, 16, 17, 26
PCON Register 17, 25, 126
$\overline{BOR}$ Bit 25
$\overline{POR}$ Bit 25
PIC16F876 Pinout Description 7
PICDEM-1 Low-Cost PICmicro Demo Board 147
PICDEM-2 Low-Cost PIC16CXX Demo Board 147
PICDEM-3 Low-Cost PIC16CXXX Demo Board 147
PICSTART® Plus Entry Level Development System 147
PIE1 Register 17, 21
PIE2 Register 17, 23
Pinout Descriptions
PIC16F873/PIC16F876 7
PIC16F874/PIC16F877 8
PIR1 Register 22
PIR2 Register 24
POP 26
PORTA 7, 8, 17
Analog Port Pins 7, 8
Initialization 29
PORTA Register 29
RA3, RA0 and RA5 Port Pins 29
RA4/T0CKI Pin 7, 8, 29
RA5/$\overline{SS}$/AN4 Pin 7, 8
TRISA Register 29
PORTA Register 15
PORTB 7, 8, 17
PORTB Register 31
Pull-up Enable ($\overline{RBPU}$ Bit) 19
RB0/INT Edge Select (INTEDG Bit) 19
RB0/INT Pin, External 7, 8, 132
RB3:RB0 Port Pins 31
RB7:RB4 Interrupt on Change 132
RB7:RB4 Interrupt on Change Enable (RBIE Bit) 20, 132
RB7:RB4 Interrupt on Change Flag (RBIF Bit) 20, 31, 132
RB7:RB4 Port Pins 31
TRISB Register 31
PORTB Register 15
PORTC 7, 8, 17
Block Diagram 33
PORTC Register 33
RC0/T1OSO/T1CKI Pin 7, 8
RC1/T1OSI/CCP2 Pin 7, 8
RC2/CCP1 Pin 7, 8
RC3/SCK/SCL Pin 7, 8
RC4/SDI/SDA Pin 7, 8
RC5/SDO Pin 7, 8
RC6/TX/CK Pin 7, 8, 96
RC7/RX/DT Pin 7, 8, 96, 97
TRISC Register 33, 95
PORTC Register 15
PORTD 9, 17, 38
Block Diagram 35
Parallel Slave Port (PSP) Function 35
PORTD Register 35
TRISD Register 35
PORTD Register 15
PORTE 9, 17
Analog Port Pins 9, 37, 38
Block Diagram 36
Input Buffer Full Status (IBF Bit) 36
Input Buffer Overflow (IBOV Bit) 36
Output Buffer Full Status (OBF Bit) 36
PORTE Register 36

PSP Mode Select (PSPMODE Bit) 35, 36, 38
RE0/RD/AN5 Pin 9, 37, 38
RE1/WR/AN6 Pin 9, 37, 38
RE2/CS/AN7 Pin 9, 37, 38
TRISE Register 36
PORTE Register 15
Postscaler, WDT
Assignment (PSA Bit) 19
Rate Select (PS2:PS0 Bits) 19
Power-on Reset (POR) 121, 125, 126, 127, 128
Oscillator Start-up Timer (OST) 121, 126
POR Status (POR Bit) 25
Power Control (PCON) Register 126
Power-down (PD Bit) 18, 125
Power-up Timer (PWRT) 121, 126
Time-out (TO Bit) 18, 125
Time-out Sequence on Power-up 129, 130
PR2 17
PR2 Register 16, 55
Prescaler, Timer0
Assignment (PSA Bit) 19
Rate Select (PS2:PS0 Bits) 19
PRO MATE® II Universal Programmer 147
Product Identification System 191
Program Counter
Reset Conditions 127
Program Memory 11
Interrupt Vector 11
Paging 11, 26
Program Memory Map 11
Reset Vector 11
Program Verification 135
Programming Pin (VPP) 7, 8
Programming, Device Instructions 137
PUSH 26

R

R/W 64
R/W bit 72
R/W bit 73
RCREG 17
RCSTA Register 17, 96
CREN Bit 96
FERR Bit 96
OERR Bit 96
RX9 Bit 96
RX9D Bit 96
SPEN Bit 95, 96
SREN Bit 96
Read/Write bit, R/W 64
Receive Enable bit 66
Receive Overflow Indicator bit, SSPOV 65
Register File 12
Register File Map 13, 14
Registers
FSR Summary 17
INDF Summary 17
INTCON Summary 17
OPTION Summary 17
PCL Summary 17
PCLATH Summary 17
PORTB Summary 17
SSPSTAT 64
STATUS Summary 17
TMR0 Summary 17
TRISB Summary 17
Reset 121, 125
Block Diagram 125
Reset Conditions for All Registers 128
Reset Conditions for PCON Register 127
Reset Conditions for Program Counter 127
Reset Conditions for STATUS Register 127
Restart Condition Enabled bit 66
Revision History 183

S

SCK 67
SCL 72
SDA 72
SDI 67
SDO 67
SEEVAL® Evaluation and Programming System 148
Serial Clock, SCK 67
Serial Clock, SCL 72
Serial Data Address, SDA 72
Serial Data In, SDI 67
Serial Data Out, SDO 67
Slave Select, SS 67
SLEEP 121, 125, 134
SMP 64
Software Simulator (MPLAB-SIM) 146
SPBRG 17
SPBRG Register 16
Special Features of the CPU 121
Special Function Registers 15
Speed, Operating 1
SPI
Master Mode 68
Master Mode Timing 68
Serial Clock 67
Serial Data In 67
Serial Data Out 67
Serial Peripheral Interface (SPI) 63
Slave Mode Timing 69
Slave Mode Timing Diagram 69
Slave Select 67
SPI clock 68
SPI Mode 67
SPI Clock Edge Select, CKE 64
SPI Data Input Sample Phase Select, SMP 64
SPI Module
Slave Mode 69
SS 67
SSP 63
Block Diagram (SPI Mode) 67
RA5/SS/AN4 Pin 7, 8
RC3/SCK/SCL Pin 7, 8
RC4/SDI/SDA Pin 7, 8
RC5/SDO Pin 7, 8
SPI Mode 67
SSPADD 72
SSPBUF 68, 72
SSPCON1 65
SSPCON2 66
SSPSR 68, 72
SSPSTAT 64, 72
SSP I^2C
SSP I^2C Operation 71

© 1999 Microchip Technology Inc.

SSP Module
SPI Master Mode 68
SPI Slave Mode 69
SSPCON1 Register 71
SSP Overflow Detect bit, SSPOV 72
SSPADD Register 16, 17
SSPBUF 17, 72
SSPBUF Register 15
SSPCON Register 15
SSPCON1 ..... 65, 71
SSPCON2 66
SSPEN 65
SSPIF 22, 73
SSPM3:SSPM0 65
SSPOV 65, 72, 83
SSPSTAT 64, 72
SSPSTAT Register 16, 17
Stack 26
Overflows 26
Underflow 26
Start bit (S) 64
Start Condition Enabled bit 66
STATUS Register 17, 18
C Bit 18
DC Bit 18
IRP Bit 18
$\overline{PD}$ Bit 18, 125
RP1:RP0 Bits 18
$\overline{TO}$ Bit 18, 125
Z Bit 18
Stop bit (P) 64
Stop Condition Enable bit 66
Synchronous Serial Port 63
Synchronous Serial Port Enable bit, SSPEN 65
Synchronous Serial Port Interrupt 22
Synchronous Serial Port Mode Select bits, SSPM3:SSPM0 65

T

T1CKPS0 bit 51
T1CKPS1 bit 51
T1CON 17
T1CON Register 17, 51
T1OSCEN bit 51
$\overline{T1SYNC}$ bit 51
T2CKPS0 bit 55
T2CKPS1 bit 55
T2CON Register 17, 55
TAD 116
Timer0
Clock Source Edge Select (T0SE Bit) 19
Clock Source Select (T0CS Bit) 19
Overflow Enable (T0IE Bit) 20
Overflow Flag (T0IF Bit) 20, 132
Overflow Interrupt 132
RA4/T0CKI Pin, External Clock 7, 8
Timer1 51
RC0/T1OSO/T1CKI Pin 7, 8
RC1/T1OSI/CCP2 Pin 7, 8
Timers
Timer0
External Clock 48
Interrupt 47
Prescaler 48
Prescaler Block Diagram 47
Section 47
T0CKI 48
Timer1
Asynchronous Counter Mode 53
Capacitor Selection 53
Operation in Timer Mode 52
Oscillator 53
Prescaler 53
Resetting of Timer1 Registers 53
Resetting Timer1 using a CCP Trigger Output .. 53
Synchronized Counter Mode 52
T1CON 51
TMR1H 53
TMR1L 53
Timer2
Block Diagram 55
Postscaler 55
Prescaler 55
T2CON 55
Timing Diagrams
A/D Conversion 172
Acknowledge Sequence Timing 85
Baud Rate Generator with Clock Arbitration 78
BRG Reset Due to SDA Collision 90
Brown-out Reset 162
Bus Collision
Start Condition Timing 89
Bus Collision During a Restart Condition (Case 1) 91
Bus Collision During a Restart Condition (Case2) 91
Bus Collision During a Start Condition (SCL = 0) 90
Bus Collision During a Stop Condition 92
Bus Collision for Transmit and Acknowledge 88
Capture/Compare/PWM 164
CLKOUT and I/O 161
I^2C Bus Data 169
I^2C Bus Start/Stop bits 168
I^2C Master Mode First Start bit timing 79
I^2C Master Mode Reception timing 84
I^2C Master Mode Transmission timing 82
Master Mode Transmit Clock Arbitration 87
Power-up Timer 162
Repeat Start Condition 80
Reset 162
SPI Master Mode 68
SPI Slave Mode (CKE = 1) 69
SPI Slave Mode Timing (CKE = 0) 69
Start-up Timer 162
Stop Condition Receive or Transmit 86
Time-out Sequence on Power-up 129, 130
Timer0 163
Timer1 163
USART Asynchronous Master Transmission 100
USART Asynchronous Reception 101
USART Synchronous Receive 170
USART Synchronous Reception 107
USART Synchronous Transmission 106, 170
USART, Asynchronous Reception 104
Wake-up from SLEEP via Interrupt 135
Watchdog Timer 162

© 1999 Microchip Technology Inc.

TMR0 17
TMR0 Register 15
TMR1CS bit 51
TMR1H 17
TMR1H Register 15
TMR1L 17
TMR1L Register 15
TMR1ON bit 51
TMR2 17
TMR2 Register 15
TMR2ON bit 55
TOUTPS0 bit 55
TOUTPS1 bit 55
TOUTPS2 bit 55
TOUTPS3 bit 55
TRISA 17
TRISA Register 16
TRISB 17
TRISB Register 16
TRISC 17
TRISC Register 16
TRISD 17
TRISD Register 16
TRISE 17
TRISE Register 16, 36
IBF Bit 36
IBOV Bit 36
OBF Bit 36
PSPMODE Bit 35, 36, 38
TXREG 17
TXSTA 17
TXSTA Register 95
BRGH Bit 95
CSRC Bit 95
SYNC Bit 95
TRMT Bit 95
TX9 Bit 95
TX9D Bit 95
TXEN Bit 95

U

UA 64
Universal Synchronous Asynchronous Receiver Transmitter (USART)
Asynchronous Receiver
Setting Up Reception 103
Timing Diagram 104
Update Address, UA 64
USART 95
Asynchronous Mode 99
Receive Block Diagram 103
Asynchronous Receiver 101
Asynchronous Reception 102
Asynchronous Transmitter 99
Baud Rate Generator (BRG) 97
Baud Rate Formula 97
Baud Rates, Asynchronous Mode (BRGH=0) ... 98
High Baud Rate Select (BRGH Bit) 95
Sampling 97
Clock Source Select (CSRC Bit) 95
Continuous Receive Enable (CREN Bit) 96
Framing Error (FERR Bit) 96
Mode Select (SYNC Bit) 95
Overrun Error (OERR Bit) 96
RC6/TX/CK Pin 7, 8
RC7/RX/DT Pin 7, 8
RCSTA Register 96
Receive Block Diagram 101
Receive Data, 9th bit (RX9D Bit) 96
Receive Enable, 9-bit (RX9 Bit) 96
Serial Port Enable (SPEN Bit) 95, 96
Single Receive Enable (SREN Bit) 96
Synchronous Master Mode 105
Synchronous Master Reception 107
Synchronous Master Transmission 105
Synchronous Slave Mode 108
Transmit Block Diagram 99
Transmit Data, 9th Bit (TX9D) 95
Transmit Enable (TXEN Bit) 95
Transmit Enable, Nine-bit (TX9 Bit) 95
Transmit Shift Register Status (TRMT Bit) 95
TXSTA Register 95

W

Wake-up from SLEEP 121, 134
Interrupts 127, 128
$\overline{\text{MCLR}}$ Reset 128
Timing Diagram 135
WDT Reset 128
Watchdog Timer (WDT) 121, 133
Block Diagram 133
Enable (WDTE Bit) 133
Programming Considerations 133
RC Oscillator 133
Time-out Period 133
WDT Reset, Normal Operation 125, 127, 128
WDT Reset, SLEEP 125, 127, 128
Waveform for General Call Address Sequence 74
WCOL 65, 79, 81, 83, 85, 86
WCOL Status Flag 79
Write Collision Detect bit, WCOL 65
WWW, On-Line Support 4

© 1999 Microchip Technology Inc.

ON-LINE SUPPORT

Microchip provides on-line support on the Microchip World Wide Web (WWW) site.

The web site is used by Microchip as a means to make files and information easily available to customers. To view the site, the user must have access to the Internet and a web browser, such as Netscape or Microsoft Explorer. Files are also available for FTP download from our FTP site.

Connecting to the Microchip Internet Web Site

The Microchip web site is available by using your favorite Internet browser to attach to:

www.microchip.com

The file transfer site is available by using an FTP service to connect to:

ftp://ftp.microchip.com

The web site and file transfer site provide a variety of services. Users may download files for the latest Development Tools, Data Sheets, Application Notes, User's Guides, Articles and Sample Programs. A variety of Microchip specific business information is also available, including listings of Microchip sales offices, distributors and factory representatives. Other data available for consideration is:

- Latest Microchip Press Releases
- Technical Support Section with Frequently Asked Questions
- Design Tips
- Device Errata
- Job Postings
- Microchip Consultant Program Member Listing
- Links to other useful web sites related to Microchip Products
- Conferences for products, Development Systems, technical information and more
- Listing of seminars and events

Systems Information and Upgrade Hot Line

The Systems Information and Upgrade Line provides system users a listing of the latest versions of all of Microchip's development systems software products. Plus, this line provides information on how customers can receive any currently available upgrade kits.The Hot Line Numbers are:

1-800-755-2345 for U.S. and most of Canada, and

1-480-786-7302 for the rest of the world.

981103

Trademarks: The Microchip name, logo, PIC, PICmicro, PICSTART, PICMASTER, PRO MATE and MPLAB are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. *Flex*ROM and *fuzzy*LAB are trademarks and SQTP is a service mark of Microchip in the U.S.A.
All other trademarks mentioned herein are the property of their respective companies.

READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 786-7578.

Please list the following information, and use this outline to provide us with your comments about this Data Sheet.

To: Technical Publications Manager Total Pages Sent

RE: Reader Response

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

Telephone: (_____) _____ - _____ FAX: (_____) _____ - _____

Application (optional):

Would you like a reply? ___Y ___N

Device: **PIC16F87X** Literature Number: **DS30292B**

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this data sheet easy to follow? If not, why?

4. What additions to the data sheet do you think would enhance the structure and subject?

5. What deletions from the data sheet could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?

8. How would you improve our software, systems, and silicon products?

© 1999 Microchip Technology Inc.

PIC16F87X PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	-XX	X	/XX	XXX
Device	Frequency Range	Temperature Range	Package	Pattern

Device	PIC16F87X(1), PIC16F87XT(2) ;VDD range 4.0V to 5.5V PIC16LF87X(1), PIC16LF87XT(2) ;VDD range 2.0V to 5.5V	
Frequency Range	04 20	= 4 MHz = 20 MHz
Temperature Range	b(3) I	= 0°C to 70°C (Commercial) = -40°C to +85°C (Industrial)
Package	PQ PT SO SP P L	= MQFP (Metric PQFP) = TQFP (Thin Quad Flatpack) = SOIC = Skinny plastic dip = PDIP = PLCC
Pattern	QTP, SQTP, Code or Special Requirements (blank otherwise)	

Examples:

f) PIC16F877 -20/P 301 = Commercial temp., PDIP package, 4 MHz, normal VDD limits, QTP pattern #301.

g) PIC16F876 - 04I/SO = Industrial temp., SOIC package, 200 kHz, Extended VDD limits.

h) PIC16F877 - 04I/P = Industrial temp., PDIP package, 10MHz, normal VDD limits.

Note 1: F = CMOS FLASH
LF = Low Power CMOS FLASH
T = in tape and reel - SOIC, PLCC, MQFP, TQFP packages only.

* JW Devices are UV erasable and can be programmed to any device configuration. JW Devices meet the electrical requirement of each oscillator type (including LC devices).

Sales and Support

Data Sheets
Products supported by a preliminary Data Sheet may have an errata sheet describing minor operational differences and recommended workarounds. To determine if an errata sheet exists for a particular device, please contact one of the following:

1. Your local Microchip sales office
2. The Microchip Corporate Literature Center U.S. FAX: (480) 786-7277
3. The Microchip Worldwide Site (www.microchip.com)

Please specify which device, revision of silicon and Data Sheet (include Literature #) you are using.

New Customer Notification System
Register on our web site (www.microchip.com/cn) to receive the most current information on our products.

© 1999 Microchip Technology Inc.

NOTES:

© 1999 Microchip Technology Inc.

NOTES:

NOTES:

© 1999 Microchip Technology Inc.

NOTES:

NOTES:

© 1999 Microchip Technology Inc.

NOTES:

WORLDWIDE SALES AND SERVICE

AMERICAS

Corporate Office
Microchip Technology Inc.
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-786-7200 Fax: 480-786-7277
Technical Support: 480-786-7627
Web Address: http://www.microchip.com

Atlanta
Microchip Technology Inc.
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770-640-0034 Fax: 770-640-0307

Boston
Microchip Technology Inc.
5 Mount Royal Avenue
Marlborough, MA 01752
Tel: 508-480-9990 Fax: 508-480-8575

Chicago
Microchip Technology Inc.
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

Dallas
Microchip Technology Inc.
4570 Westgrove Drive, Suite 160
Addison, TX 75248
Tel: 972-818-7423 Fax: 972-818-2924

Dayton
Microchip Technology Inc.
Two Prestige Place, Suite 150
Miamisburg, OH 45342
Tel: 937-291-1654 Fax: 937-291-9175

Detroit
Microchip Technology Inc.
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

Los Angeles
Microchip Technology Inc.
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888 Fax: 949-263-1338

New York
Microchip Technology Inc.
150 Motor Parkway, Suite 202
Hauppauge, NY 11788
Tel: 631-273-5305 Fax: 631-273-5335

San Jose
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

AMERICAS (continued)

Toronto
Microchip Technology Inc.
5925 Airport Road, Suite 200
Mississauga, Ontario L4V 1W1, Canada
Tel: 905-405-6279 Fax: 905-405-6253

ASIA/PACIFIC

Hong Kong
Microchip Asia Pacific
Unit 2101, Tower 2
Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2-401-1200 Fax: 852-2-401-3431

Beijing
Microchip Technology, Beijing
Unit 915, 6 Chaoyangmen Bei Dajie
Dong Erhuan Road, Dongcheng District
New China Hong Kong Manhattan Building
Beijing 100027 PRC
Tel: 86-10-85282100 Fax: 86-10-85282104

India
Microchip Technology Inc.
India Liaison Office
No. 6, Legacy, Convent Road
Bangalore 560 025, India
Tel: 91-80-229-0061 Fax: 91-80-229-0062

Japan
Microchip Technology Intl. Inc.
Benex S-1 6F
3-18-20, Shinyokohama
Kohoku-Ku, Yokohama-shi
Kanagawa 222-0033 Japan
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

Korea
Microchip Technology Korea
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku
Seoul, Korea
Tel: 82-2-554-7200 Fax: 82-2-558-5934

Shanghai
Microchip Technology
RM 406 Shanghai Golden Bridge Bldg.
2077 Yan'an Road West, Hong Qiao District
Shanghai, PRC 200335
Tel: 86-21-6275-5700 Fax: 86 21-6275-5060

ASIA/PACIFIC (continued)

Singapore
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore 188980
Tel: 65-334-8870 Fax: 65-334-8850

Taiwan, R.O.C
Microchip Technology Taiwan
10F-1C 207
Tung Hua North Road
Taipei, Taiwan, ROC
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

EUROPE

United Kingdom
Arizona Microchip Technology Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5858 Fax: 44-118 921-5835

Denmark
Microchip Technology Denmark ApS
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

France
Arizona Microchip Technology SARL
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trapu
Batiment A - ler Etage
91300 Massy, France
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

Germany
Arizona Microchip Technology GmbH
Gustav-Heinemann-Ring 125
D-81739 München, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

Italy
Arizona Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-039-65791-1 Fax: 39-039-6899883

11/15/99

DNV Certification, Inc. USA — ANSI-RAB QMS ACCREDITED — DNV — DNV MSC The Netherlands Accredited by the RvA
ISO 9001 / QS-9000 REGISTERED FIRM

Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.

All rights reserved. © 1999 Microchip Technology Incorporated. Printed in the USA. 11/99 Printed on recycled paper.

Information contained in this publication regarding device applications and the like is intended for suggestion only and may be superseded by updates. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights. The Microchip logo and name are registered trademarks of Microchip Technology Inc. in the U.S.A. and other countries. All rights reserved. All other trademarks mentioned herein are the property of their respective companies.

Anexo B

Especificaciones técnicas del XR2206

XR-2206

Monolithic
Function Generator

...the analog plus company™

June 1997-3

FEATURES

- Low-Sine Wave Distortion, 0.5%, Typical
- Excellent Temperature Stability, 20ppm/°C, Typ.
- Wide Sweep Range, 2000:1, Typical
- Low-Supply Sensitivity, 0.01%V, Typ.
- Linear Amplitude Modulation
- TTL Compatible FSK Controls
- Wide Supply Range, 10V to 26V
- Adjustable Duty Cycle, 1% TO 99%

APPLICATIONS

- Waveform Generation
- Sweep Generation
- AM/FM Generation
- V/F Conversion
- FSK Generation
- Phase-Locked Loops (VCO)

GENERAL DESCRIPTION

The XR-2206 is a monolithic function generator integrated circuit capable of producing high quality sine, square, triangle, ramp, and pulse waveforms of high-stability and accuracy. The output waveforms can be both amplitude and frequency modulated by an external voltage. Frequency of operation can be selected externally over a range of 0.01Hz to more than 1MHz.

The circuit is ideally suited for communications, instrumentation, and function generator applications requiring sinusoidal tone, AM, FM, or FSK generation. It has a typical drift specification of 20ppm/°C. The oscillator frequency can be linearly swept over a 2000:1 frequency range with an external control voltage, while maintaining low distortion.

ORDERING INFORMATION

Part No.	Package	Operating Temperature Range
XR-2206M	16 Lead 300 Mil CDIP	-55°C to +125°C
XR-2206P	16 Lead 300 Mil PDIP	–40°C to +85°C
XR-2206CP	16 Lead 300 Mil PDIP	0°C to +70°C
XR-2206D	16 Lead 300 Mil JEDEC SOIC	0°C to +70°C

©1972
EXAR Corporation, 48720 Kato Road, Fremont, CA 94538 ◆ (510) 668-7000 ◆ (510) 668-7017

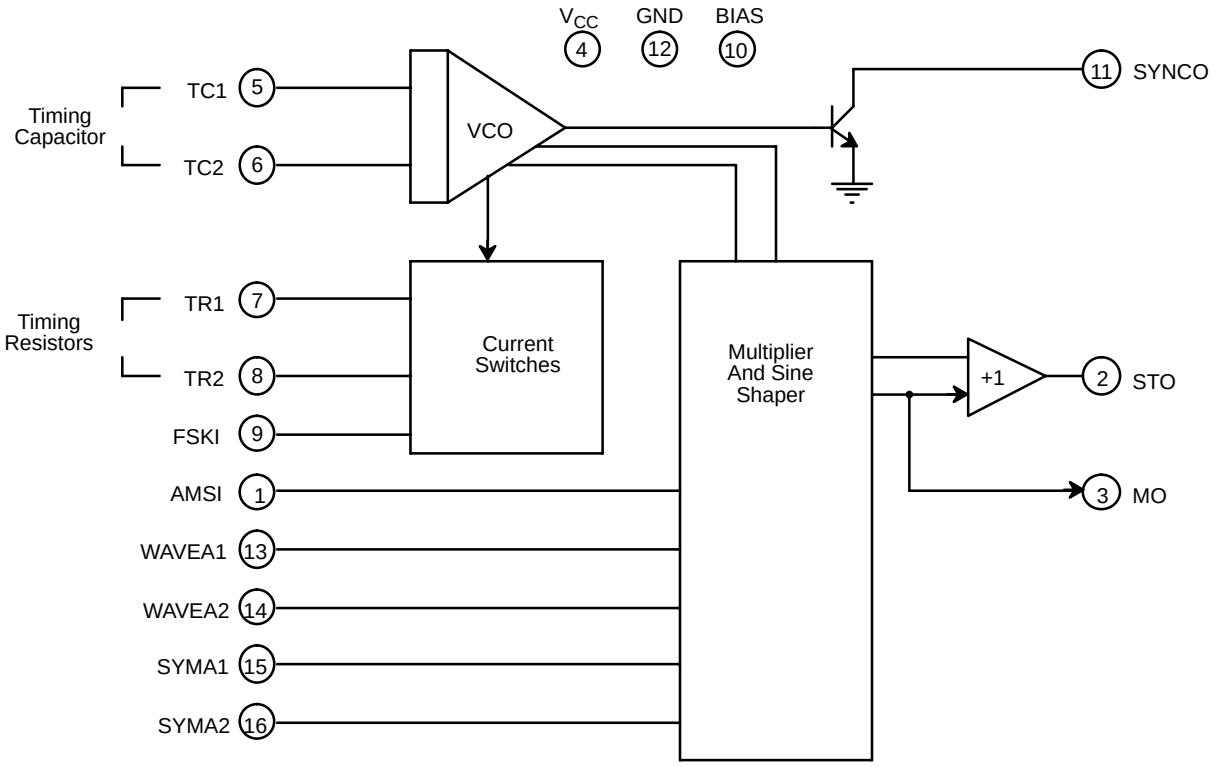
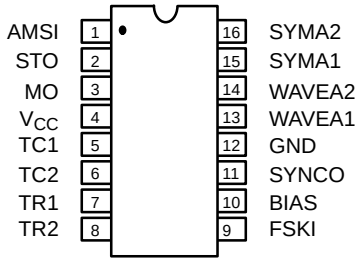
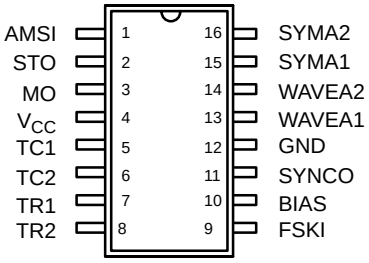


Figure 1. XR-2206 Block Diagram



16 Lead PDIP, CDIP (0.300")



16 Lead SOIC (Jedec, 0.300")

PIN DESCRIPTION

Pin #	Symbol	Type	Description
1	AMSI	I	**Amplitude Modulating Signal Input.**
2	STO	O	**Sine or Triangle Wave Output.**
3	MO	O	**Multiplier Output.**
4	V_{CC}		**Positive Power Supply.**
5	TC1	I	**Timing Capacitor Input.**
6	TC2	I	**Timing Capacitor Input.**
7	TR1	O	**Timing Resistor 1 Output.**
8	TR2	O	**Timing Resistor 2 Output.**
9	FSKI	I	**Frequency Shift Keying Input.**
10	BIAS	O	**Internal Voltage Reference.**
11	SYNCO	O	**Sync Output.** This output is a open collector and needs a pull up resistor to V_{CC}.
12	GND		**Ground pin.**
13	WAVEA1	I	**Wave Form Adjust Input 1.**
14	WAVEA2	I	**Wave Form Adjust Input 2.**
15	SYMA1	I	**Wave Symetry Adjust 1.**
16	SYMA2	I	**Wave Symetry Adjust 2.**

DC ELECTRICAL CHARACTERISTICS

Test Conditions: Test Circuit of *Figure 2* **Vcc = 12V, T_A = 25°C, C = 0.01μF, R_1 = 100kΩ, R_2 = 10kΩ, R_3 = 25kΩ** Unless Otherwise Specified. S_1 open for triangle, closed for sine wave.

Parameters	XR-2206M/P			XR-2206CP/D			Units	Conditions
	Min.	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**		
General Characteristics								
Single Supply Voltage	**10**		**26**	10		26	V	
Split-Supply Voltage	**±5**		**±13**	±5		±13	V	
Supply Current		12	**17**		14	20	mA	$R_1 \geq 10k\Omega$
Oscillator Section								
Max. Operating Frequency	**0.5**	1		0.5	1		MHz	C = 1000pF, R_1 = 1kΩ
Lowest Practical Frequency		0.01			0.01		Hz	C = 50μF, R_1 = 2MΩ
Frequency Accuracy		±1	**±4**		±2		% of f_o	$f_o = 1/R_1C$
Temperature Stability Frequency		±10	**±50**		±20		ppm/°C	0°C ≤ T_A ≤ 70°C $R_1 = R_2$ = 20kΩ
Sine Wave Amplitude Stability[2]		4800			4800		ppm/°C	
Supply Sensitivity		0.01	**0.1**		0.01		%/V	V_{LOW} = 10V, V_{HIGH} = 20V, $R_1 = R_2$ = 20kΩ
Sweep Range	1000:1	2000:1			2000:1		$f_H = f_L$	f_H @ R_1 = 1kΩ f_L @ R_1 = 2MΩ
Sweep Linearity								
10:1 Sweep		2			2		%	f_L = 1kHz, f_H = 10kHz
1000:1 Sweep		8			8		%	f_L = 100Hz, f_H = 100kHz
FM Distortion		0.1			0.1		%	±10% Deviation
Recommended Timing Components								
Timing Capacitor: C	**0.001**		100	0.001		100	μF	*Figure 5*
Timing Resistors: R_1 & R_2	**1**		2000	1		2000	kΩ	
Triangle Sine Wave Output[1]								*Figure 3*
Triangle Amplitude		160			160		mV/kΩ	*Figure 2,* S_1 Open
Sine Wave Amplitude	**40**	60	80		60		mV/kΩ	*Figure 2*, S_1 Closed
Max. Output Swing		6			6		Vp-p	
Output Impedance		600			600		Ω	
Triangle Linearity		1			1		%	
Amplitude Stability		0.5			0.5		dB	For 1000:1 Sweep
Sine Wave Distortion								
Without Adjustment		2.5			2.5		%	R_1 = 30kΩ
With Adjustment		0.4	**1.0**		0.5	1.5	%	See *Figure 7* and *Figure 8*

Notes

[1] Output amplitude is directly proportional to the resistance, R_3, on Pin 3. See Figure 3.

[2] For maximum amplitude stability, R_3 should be a positive temperature coefficient resistor.

***Bold face parameters** are covered by production test and guaranteed over operating temperature range.*

DC ELECTRICAL CHARACTERISTICS (CONT'D)

Parameters	XR-2206M/P			XR-2206CP/D			Units	Conditions
	Min.	Typ.	Max.	Min.	Typ.	Max.		
Amplitude Modulation								
Input Impedance	50	100		50	100		kΩ	
Modulation Range		100			100		%	
Carrier Suppression		55			55		dB	
Linearity		2			2		%	For 95% modulation
Square-Wave Output								
Amplitude		12			12		Vp-p	Measured at Pin 11.
Rise Time		250			250		ns	C_L = 10pF
Fall Time		50			50		ns	C_L = 10pF
Saturation Voltage		0.2	**0.4**		0.2	0.6	V	I_L = 2mA
Leakage Current		0.1	**20**		0.1	100	μA	V_{CC} = 26V
FSK Keying Level (Pin 9)	0.8	1.4	**2.4**	0.8	1.4	2.4	V	See section on circuit controls
Reference Bypass Voltage	2.9	3.1	**3.3**	2.5	3	3.5	V	Measured at Pin 10.

Notes
[1] Output amplitude is directly proportional to the resistance, R_3, on Pin 3. See Figure 3.
[2] For maximum amplitude stability, R_3 should be a positive temperature coefficient resistor.
***Bold face parameters** are covered by production test and guaranteed over operating temperature range.*

Specifications are subject to change without notice

ABSOLUTE MAXIMUM RATINGS

Power Supply 26V
Power Dissipation 750mW
Derate Above 25°C . . . . . 5mW/°C
Total Timing Current 6mA
Storage Temperature . . . . . -65°C to +150°C

SYSTEM DESCRIPTION

The XR-2206 is comprised of four functional blocks; a voltage-controlled oscillator (VCO), an analog multiplier and sine-shaper; a unity gain buffer amplifier; and a set of current switches.

The VCO produces an output frequency proportional to an input current, which is set by a resistor from the timing terminals to ground. With two timing pins, two discrete output frequencies can be independently produced for FSK generation applications by using the FSK input control pin. This input controls the current switches which select one of the timing resistor currents, and routes it to the VCO.

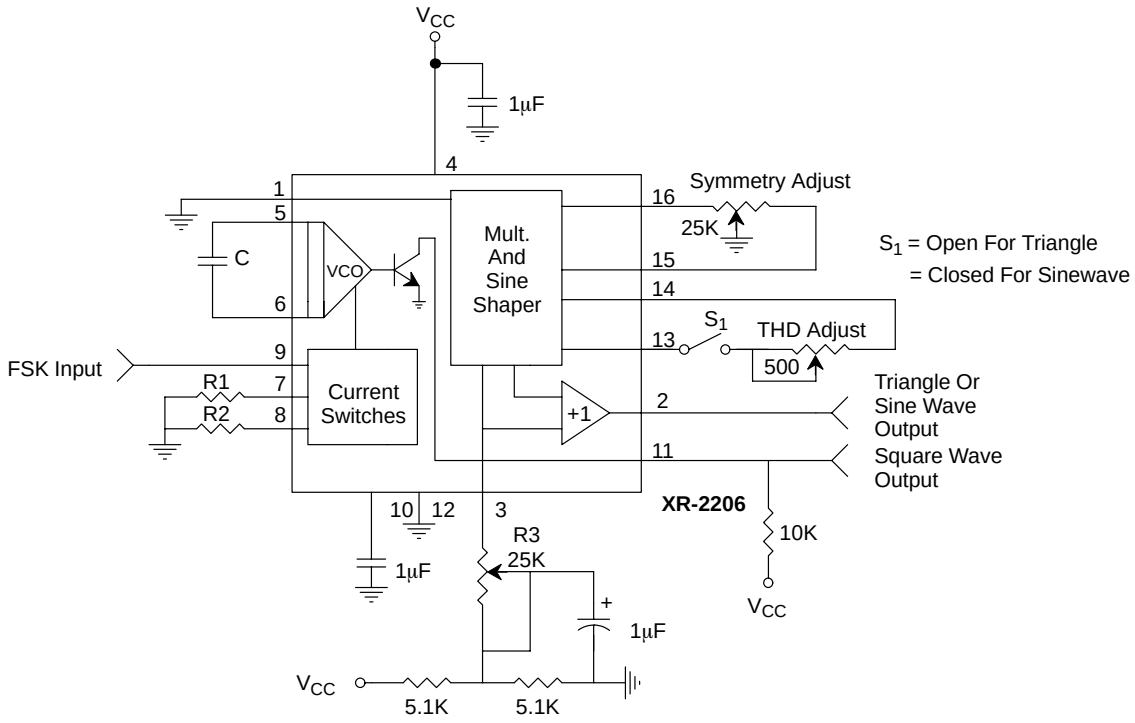


Figure 2. Basic Test Circuit

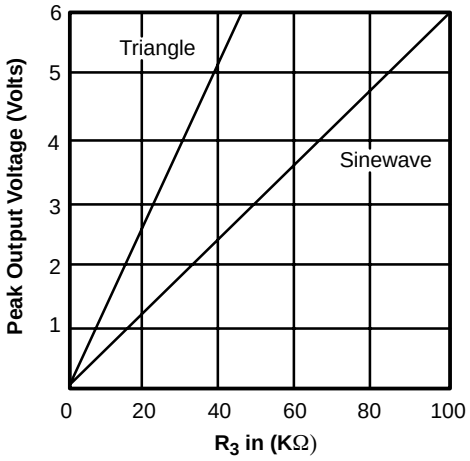


Figure 3. Output Amplitude as a Function of the Resistor, R3, at Pin 3

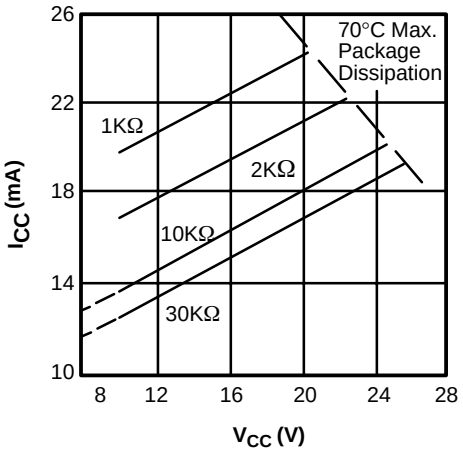


Figure 4. Supply Current vs Supply Voltage, Timing, R

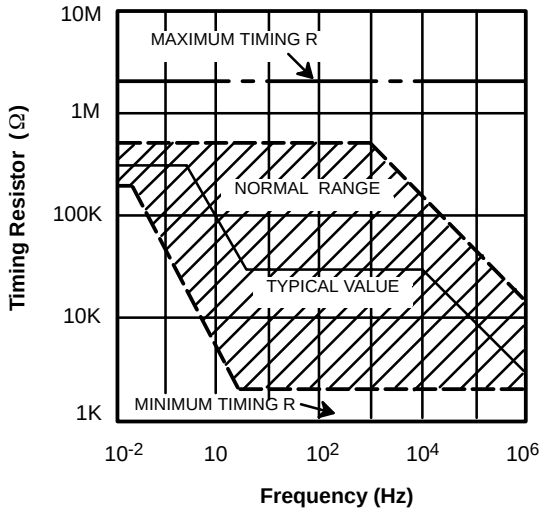


Figure 5. R versus Oscillation Frequency.

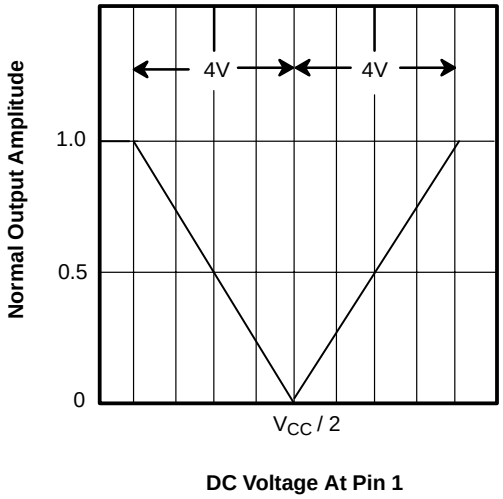


Figure 6. Normalized Output Amplitude versus DC Bias at AM Input (Pin 1)

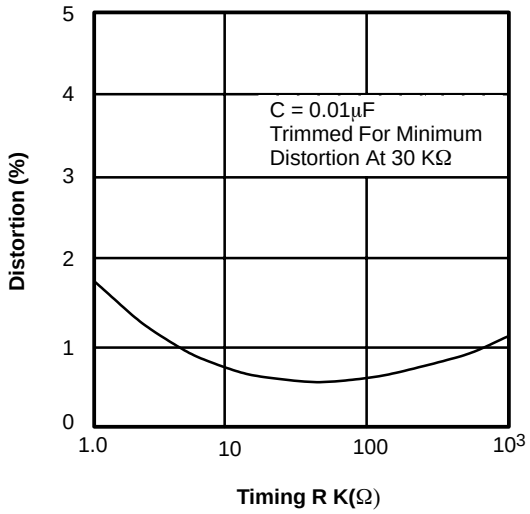


Figure 7. Trimmed Distortion versus Timing Resistor.

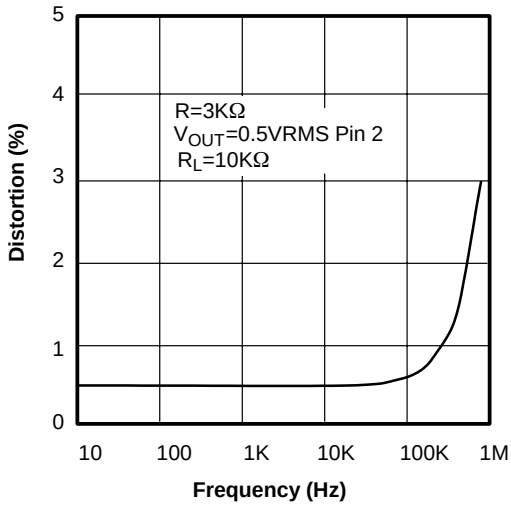


Figure 8. Sine Wave Distortion versus Operating Frequency with Timing Capacitors Varied.

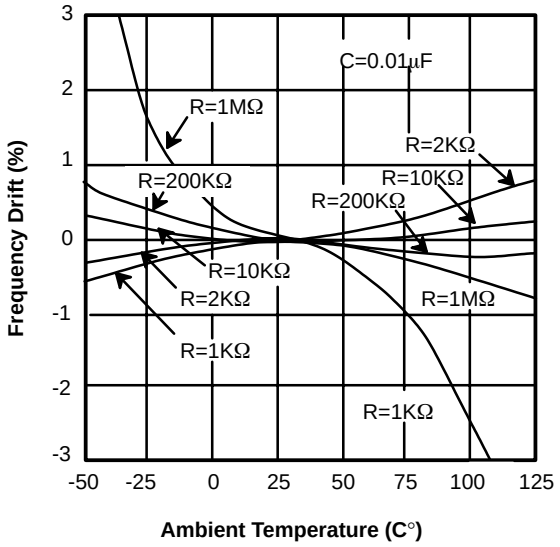


Figure 9. Frequency Drift versus Temperature.

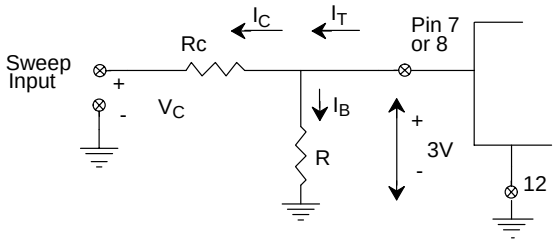


Figure 10. Circuit Connection for Frequency Sweep.

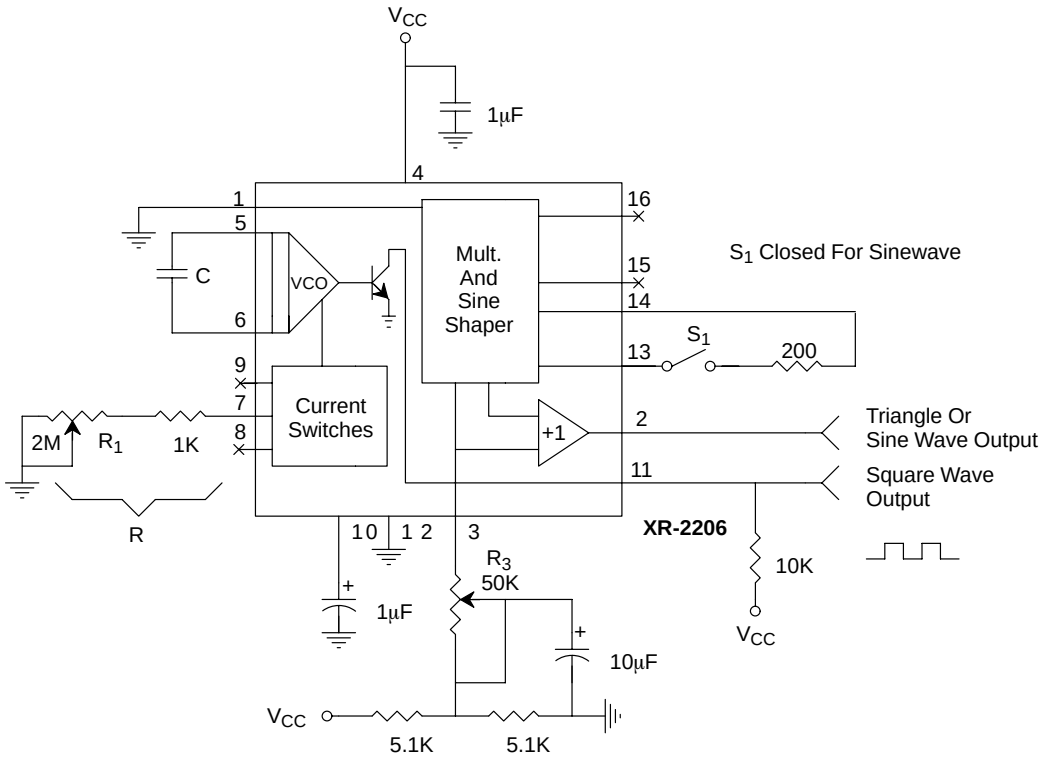


Figure 11. Circuit tor Sine Wave Generation without External Adjustment. (See *Figure 3* for Choice of R_3)

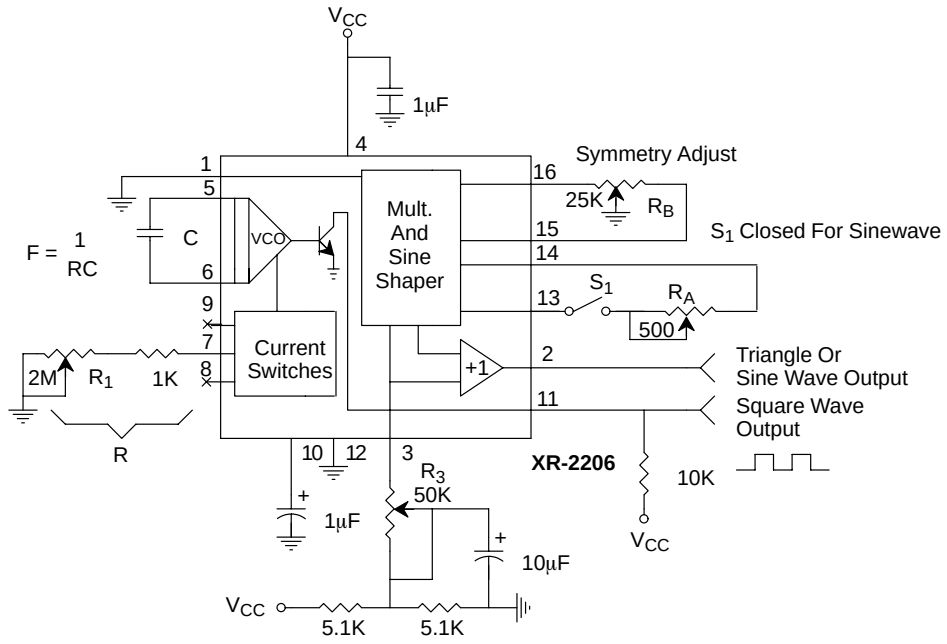


Figure 12. Circuit for Sine Wave Generation with Minimum Harmonic Distortion. (R_3 Determines Output Swing - See *Figure 3*)

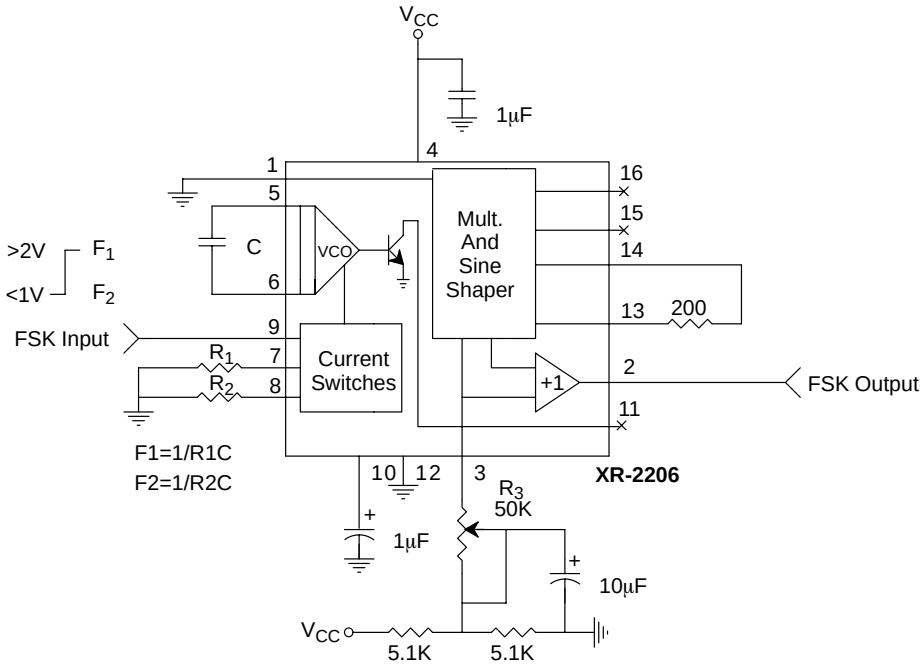


Figure 13. Sinusoidal FSK Generator

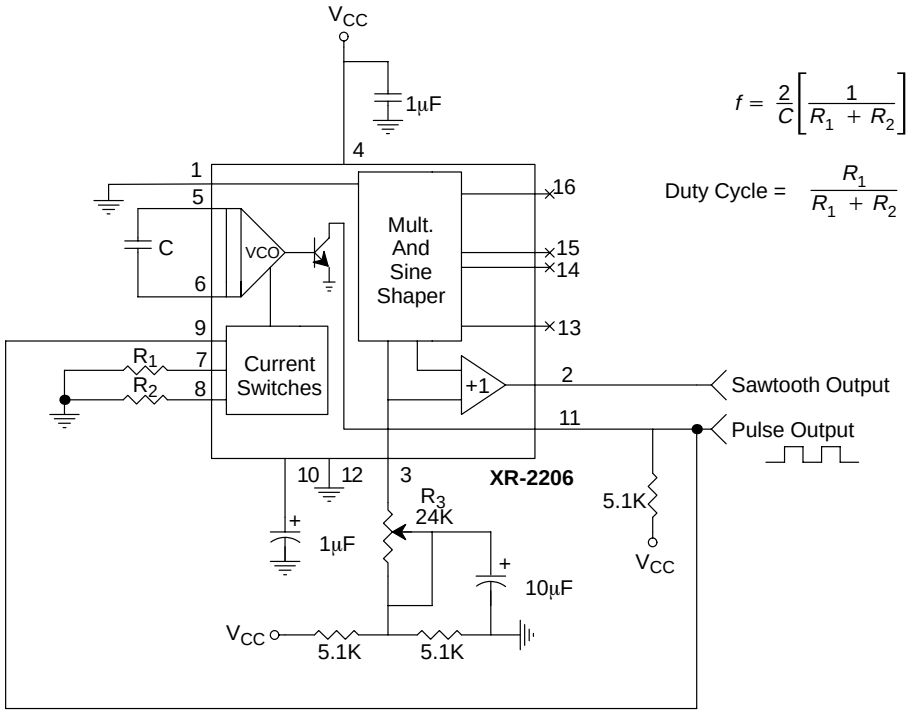


Figure 14. Circuit for Pulse and Ramp Generation.

Frequency-Shift Keying

The XR-2206 can be operated with two separate timing resistors, R_1 and R_2, connected to the timing Pin 7 and 8, respectively, as shown in *Figure 13.* Depending on the polarity of the logic signal at Pin 9, either one or the other of these timing resistors is activated. If Pin 9 is open-circuited or connected to a bias voltage $\geq$ 2V, only R_1 is activated. Similarly, if the voltage level at Pin 9 is $\leq$1V, only R_2 is activated. Thus, the output frequency can be keyed between two levels. f_1 and f_2, as:

$f_1 = 1/R_1C$ and $f_2 = 1/R_2C$

For split-supply operation, the keying voltage at Pin 9 is referenced to V^-.

Output DC Level Control

The dc level at the output (Pin 2) is approximately the same as the dc bias at Pin 3. In *Figure 11*, *Figure 12* and *Figure 13*, Pin 3 is biased midway between V+ and ground, to give an output dc level of $\approx V^+/2$.

APPLICATIONS INFORMATION

Sine Wave Generation

Without External Adjustment

Figure 11 shows the circuit connection for generating a sinusoidal output from the XR-2206. The potentiometer, R_1 at Pin 7, provides the desired frequency tuning. The maximum output swing is greater than $V^+/2$, and the typical distortion (THD) is < 2.5%. If lower sine wave distortion is desired, additional adjustments can be provided as described in the following section.

The circuit of *Figure 11* can be converted to split-supply operation, simply by replacing all ground connections with V^-. For split-supply operation, R_3 can be directly connected to ground.

With External Adjustment:

The harmonic content of sinusoidal output can be reduced to -0.5% by additional adjustments as shown in *Figure 12.* The potentiometer, R_A, adjusts the sine-shaping resistor, and R_B provides the fine adjustment for the waveform symmetry. The adjustment procedure is as follows:

1. Set R_B at midpoint and adjust R_A for minimum distortion.
2. With R_A set as above, adjust R_B to further reduce distortion.

Triangle Wave Generation

The circuits of *Figure 11* and *Figure 12* can be converted to triangle wave generation, by simply open-circuiting Pin 13 and 14 (i.e., S_1 open). Amplitude of the triangle is approximately twice the sine wave output.

FSK Generation

Figure 13 shows the circuit connection for sinusoidal FSK signal operation. Mark and space frequencies can be independently adjusted by the choice of timing resistors, R_1 and R_2; the output is phase-continuous during transitions. The keying signal is applied to Pin 9. The circuit can be converted to split-supply operation by simply replacing ground with V^-.

Pulse and Ramp Generation

Figure 14 shows the circuit for pulse and ramp waveform generation. In this mode of operation, the FSK keying terminal (Pin 9) is shorted to the square-wave output (Pin 11), and the circuit automatically frequency-shift keys itself between two separate frequencies during the positive-going and negative-going output waveforms. The pulse width and duty cycle can be adjusted from 1% to 99% by the choice of R_1 and R_2. The values of R_1 and R_2 should be in the range of 1kΩ to 2MΩ.

PRINCIPLES OF OPERATION

Description of Controls

Frequency of Operation:

The frequency of oscillation, f_0, is determined by the external timing capacitor, C, across Pin 5 and 6, and by the timing resistor, R, connected to either Pin 7 or 8. The frequency is given as:

$$f_0 = \frac{1}{RC}\ Hz$$

and can be adjusted by varying either R or C. The recommended values of R, for a given frequency range, as shown in *Figure 5.* Temperature stability is optimum for 4kΩ < R < 200kΩ. Recommended values of C are from 1000pF to 100μF.

Frequency Sweep and Modulation:

Frequency of oscillation is proportional to the total timing current, I_T, drawn from Pin 7 or 8:

$$f = \frac{320 I_T(mA)}{C(\mu F)}\ Hz$$

Timing terminals (Pin 7 or 8) are low-impedance points, and are internally biased at +3V, with respect to Pin 12. Frequency varies linearly with IT, over a wide range of current values, from 1μA to 3mA. The frequency can be controlled by applying a control voltage, V_C, to the activated timing pin as shown in *Figure 10.* The frequency of oscillation is related to VC as:

$$f = \frac{1}{RC}\left(1 + \frac{R}{R_C}\left(1 - \frac{V_C}{3}\right)\right) Hz$$

where V_C is in volts. The voltage-to-frequency conversion gain, K, is given as:

$$K = \partial f / \partial V_C = -\frac{0.32}{R_C C}\ \ Hz/V$$

CAUTION: *For safety operation of the circuit, I_T should be limited to ≤ 3mA.*

Output Amplitude:

Maximum output amplitude is inversely proportional to the external resistor, R_3, connected to Pin 3 (see *Figure 3*). For sine wave output, amplitude is approximately 60mV peak per kΩ of R_3; for triangle, the peak amplitude is approximately 160mV peak per kΩ of R_3. Thus, for example, R_3 = 50kΩ would produce approximately 13V sinusoidal output amplitude.

Amplitude Modulation:

Output amplitude can be modulated by applying a dc bias and a modulating signal to Pin 1. The internal impedance at Pin 1 is approximately 100kΩ. Output amplitude varies linearly with the applied voltage at Pin 1, for values of dc bias at this pin, within 14 volts of $V_{CC}/2$ as shown in *Figure 6.* As this bias level approaches $V_{CC}/2$, the phase of the output signal is reversed, and the amplitude goes through zero. This property is suitable for phase-shift keying and suppressed-carrier AM generation. Total dynamic range of amplitude modulation is approximately 55dB.

CAUTION: *AM control must be used in conjunction with a well-regulated supply, since the output amplitude now becomes a function of* V_{CC}.

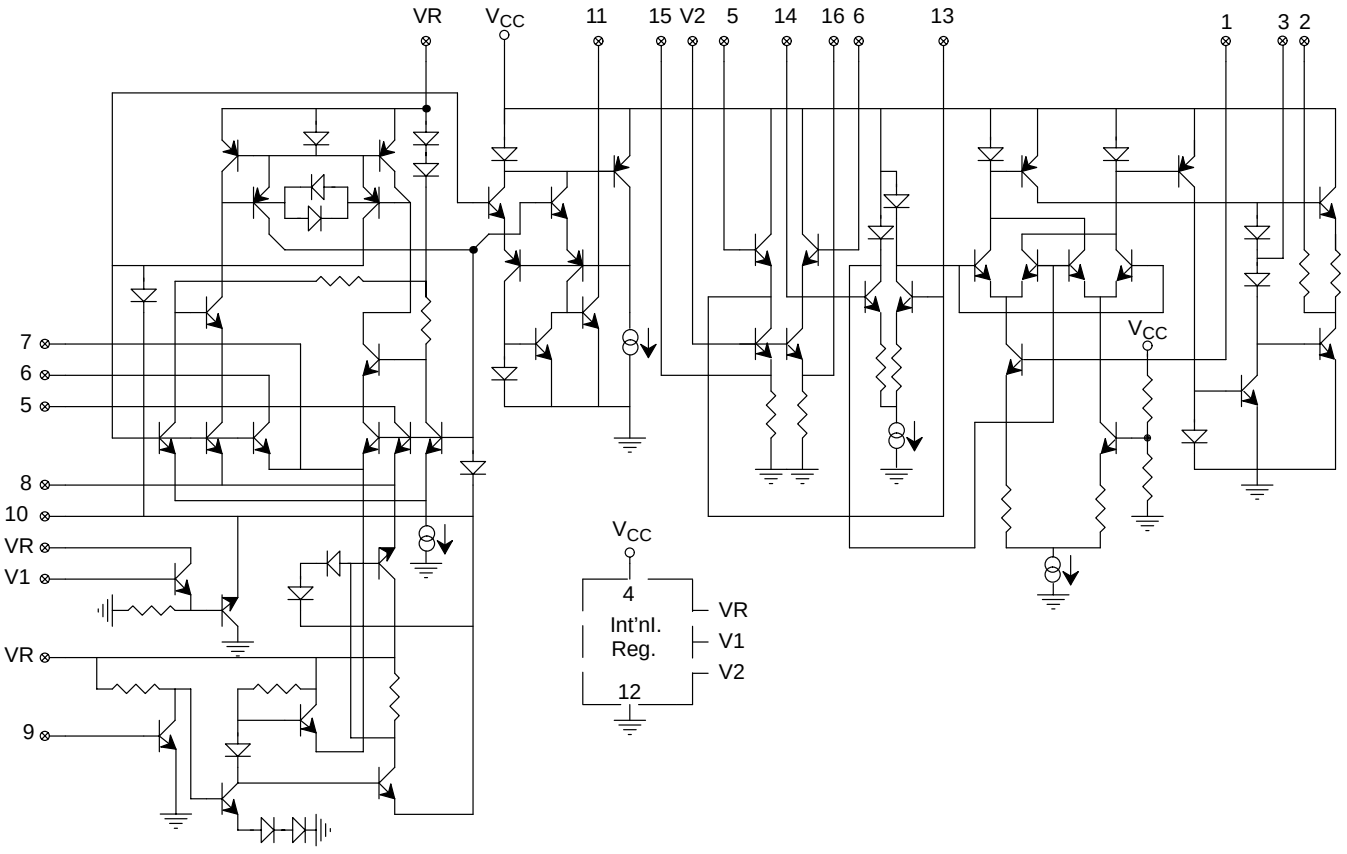
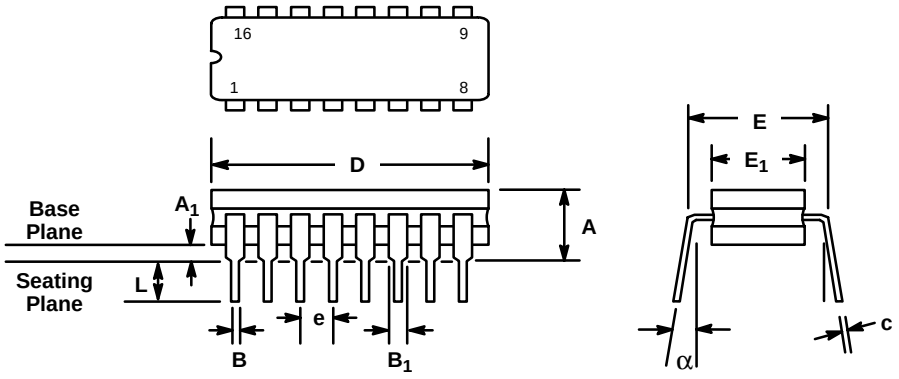


Figure 15. Equivalent Schematic Diagram

16 LEAD CERAMIC DUAL-IN-LINE (300 MIL CDIP)

Rev. 1.00

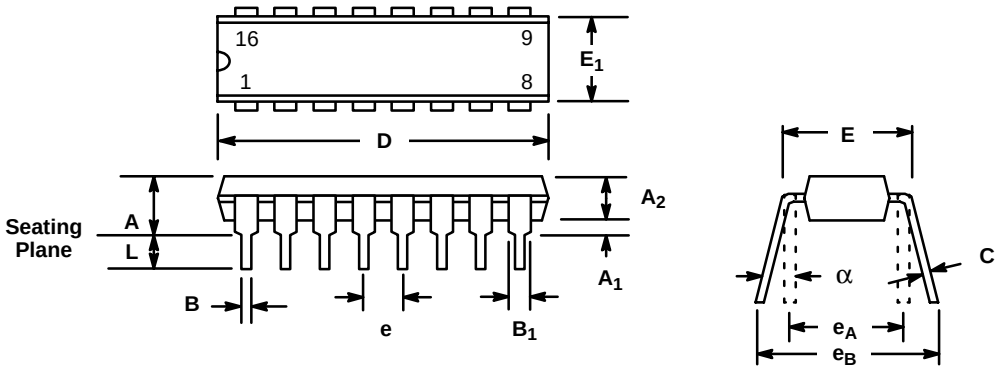


SYMBOL	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.100	0.200	2.54	5.08
A_1	0.015	0.060	0.38	1.52
B	0.014	0.026	0.36	0.66
B_1	0.045	0.065	1.14	1.65
c	0.008	0.018	0.20	0.46
D	0.740	0.840	18.80	21.34
E_1	0.250	0.310	6.35	7.87
E	0.300 BSC		7.62 BSC	
e	0.100 BSC		2.54 BSC	
L	0.125	0.200	3.18	5.08
α	0°	15°	0°	15°

Note: The control dimension is the inch column

16 LEAD PLASTIC DUAL-IN-LINE (300 MIL PDIP)

Rev. 1.00

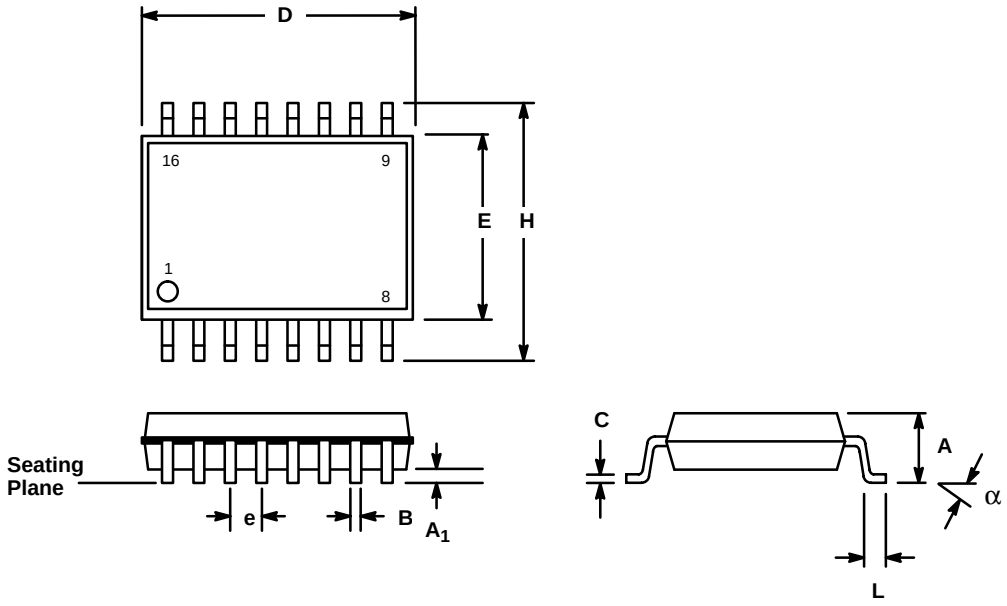


SYMBOL	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.145	0.210	3.68	5.33
A_1	0.015	0.070	0.38	1.78
A_2	0.115	0.195	2.92	4.95
B	0.014	0.024	0.36	0.56
B_1	0.030	0.070	0.76	1.78
C	0.008	0.014	0.20	0.38
D	0.745	0.840	18.92	21.34
E	0.300	0.325	7.62	8.26
E_1	0.240	0.280	6.10	7.11
e	0.100 BSC		2.54 BSC	
e_A	0.300 BSC		7.62 BSC	
e_B	0.310	0.430	7.87	10.92
L	0.115	0.160	2.92	4.06
α	0°	15°	0°	15°

Note: The control dimension is the inch column

16 LEAD SMALL OUTLINE (300 MIL JEDEC SOIC)

Rev. 1.00



SYMBOL	INCHES MIN	INCHES MAX	MILLIMETERS MIN	MILLIMETERS MAX
A	0.093	0.104	2.35	2.65
A_1	0.004	0.012	0.10	0.30
B	0.013	0.020	0.33	0.51
C	0.009	0.013	0.23	0.32
D	0.398	0.413	10.10	10.50
E	0.291	0.299	7.40	7.60
e	0.050 BSC		1.27 BSC	
H	0.394	0.419	10.00	10.65
L	0.016	0.050	0.40	1.27
α	0°	8°	0°	8°

Note: The control dimension is the millimeter column

NOTICE

EXAR Corporation reserves the right to make changes to the products contained in this publication in order to improve design, performance or reliability. EXAR Corporation assumes no responsibility for the use of any circuits described herein, conveys no license under any patent or other right, and makes no representation that the circuits are free of patent infringement. Charts and schedules contained here in are only for illustration purposes and may vary depending upon a user's specific application. While the information in this publication has been carefully checked; no responsibility, however, is assumed for inaccuracies.

EXAR Corporation does not recommend the use of any of its products in life support applications where the failure or malfunction of the product can reasonably be expected to cause failure of the life support system or to significantly affect its safety or effectiveness. Products are not authorized for use in such applications unless EXAR Corporation receives, in writing, assurances to its satisfaction that: (a) the risk of injury or damage has been minimized; (b) the user assumes all such risks; (c) potential liability of EXAR Corporation is adequately protected under the circumstances.

Copyright 1972 EXAR Corporation
Datasheet June 1997
Reproduction, in part or whole, without the prior written consent of EXAR Corporation is prohibited.

Anexo C

Especificaciones técnicas del XR2211

XR-2211

FSK Demodulator/
Tone Decoder

...the analog plus company™

June 1997-3

FEATURES

- Wide Frequency Range, 0.01Hz to 300kHz
- Wide Supply Voltage Range, 4.5V to 20V
- HCMOS/TTL/Logic Compatibility
- FSK Demodulation, with Carrier Detection
- Wide Dynamic Range, 10mV to 3V rms
- Adjustable Tracking Range, ±1% to 80%
- Excellent Temp. Stability, ±50ppm/°C, max.

APPLICATIONS

- Caller Identification Delivery
- FSK Demodulation
- Data Synchronization
- Tone Decoding
- FM Detection
- Carrier Detection

GENERAL DESCRIPTION

The XR-2211 is a monolithic phase-locked loop (PLL) system especially designed for data communications applications. It is particularly suited for FSK modem applications. It operates over a wide supply voltage range of 4.5 to 20V and a wide frequency range of 0.01Hz to 300kHz. It can accommodate analog signals between 10mV and 3V, and can interface with conventional DTL, TTL, and ECL logic families. The circuit consists of a basic PLL for tracking an input signal within the pass band, a quadrature phase detector which provides carrier detection, and an FSK voltage comparator which provides FSK demodulation. External components are used to independently set center frequency, bandwidth, and output delay. An internal voltage reference proportional to the power supply is provided at an output pin.

The XR-2211 is available in 14 pin packages specified for military and industrial temperature ranges.

ORDERING INFORMATION

Part No.	Package	Operating Temperature Range
XR-2211M	14 Pin CDIP (0.300”)	-55°C to +125°C
XR-2211N	14 Pin CDIP (0.300”)	-40°C to +85°C
XR-2211P	14 Pin PDIP (0.300”)	-40°C to +85°C
XR-2211ID	14 Lead SOIC (Jedec, 0.150”)	-40°C to +85°C

©1992

EXAR Corporation, 48720 Kato Road, Fremont, CA 94538 ◆ (510) 668-7000 ◆ FAX (510) 668-7017

BLOCK DIAGRAM

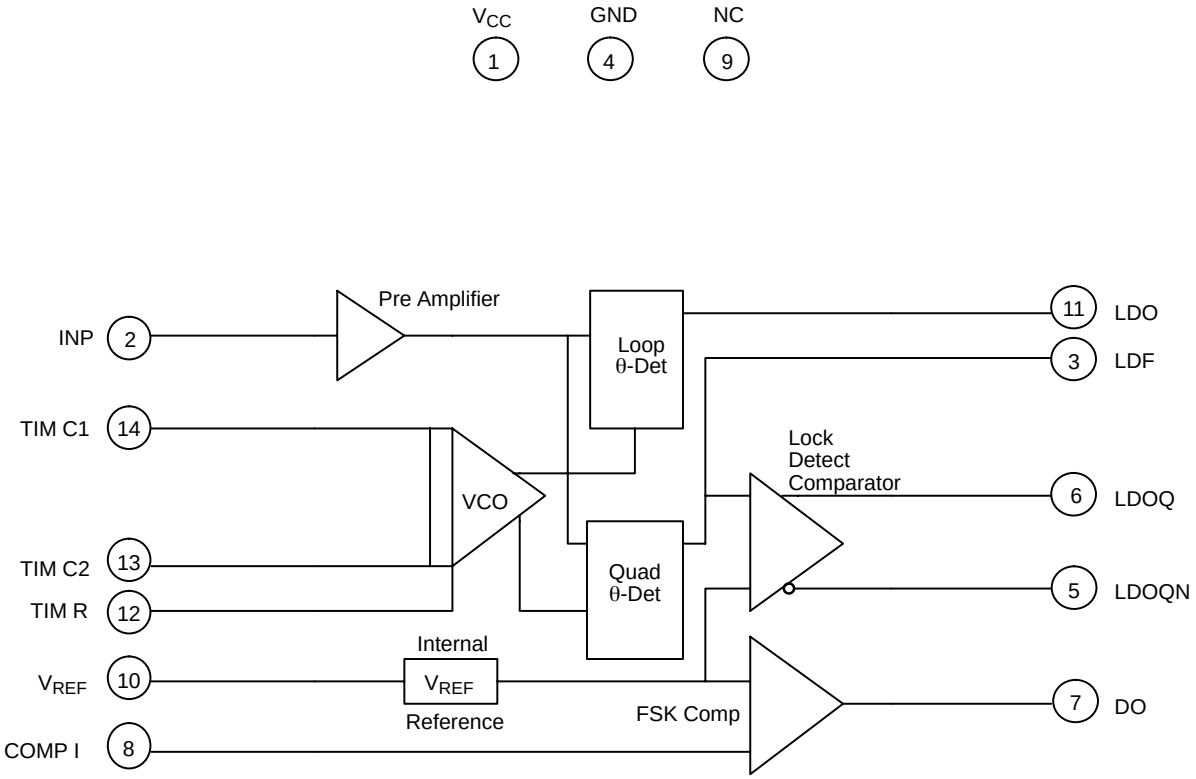
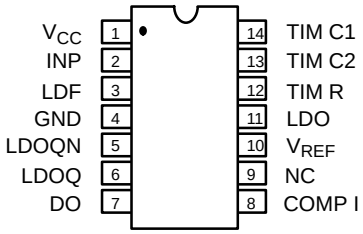
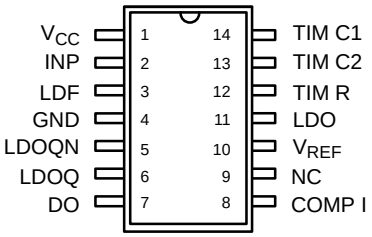


Figure 1. XR-2211 Block Diagram

PIN CONFIGURATION



14 Lead CDIP, PDIP (0.300")



14 Lead SOIC (Jedec, 0.150")

PIN DESCRIPTION

Pin #	Symbol	Type	Description
1	V_{CC}		**Positive Power Supply.**
2	INP	I	**Receive Analog Input.**
3	LDF	O	**Lock Detect Filter.**
4	GND		**Ground Pin.**
5	LDOQN	O	**Lock Detect Output Not.** This output will be low if the VCO is in the capture range.
6	LDOQ	O	**Lock Detect Output.** This output will be high if the VCO is in the capture range.
7	DO	O	**Data Output.** Decoded FSK output.
8	COMP I	I	**FSK Comparator Input.**
9	NC		**Not Connected.**
10	V_{REF}	O	**Internal Voltage Reference.** The value of V_{REF} is V_{CC}/2 - 650mV.
11	LDO	O	**Loop Detect Output.** This output provides the result of the quadrature phase detection.
12	TIM R	I	**Timing Resistor Input.** This pin connects to the timing resistor of the VCO.
13	TIM C2	I	**Timing Capacitor Input.** The timing capacitor connects between this pin and pin 14.
14	TIM C1	I	**Timing Capacitor Input.** The timing capacitor connects between this pin and pin 13.

ELECTRICAL CHARACTERISTICS

Test Conditions: V_{CC} = 12V, T_A = +25°C, R_O = 30KΩ, C_O = 0.033μF, unless otherwise specified.

Parameter	Min.	Typ.	Max.	Unit	Conditions
General					
Supply Voltage	**4.5**		**20**	V	
Supply Current		4	**7**	mA	$R_0 \geq$ 10KΩ. See *Figure 4.*
Oscillator Section					
Frequency Accuracy		±1	**±3**	%	Deviation from $f_O = 1/R_0 C_0$
Frequency Stability					
Temperature		±20	±50	ppm/°C	See *Figure 8.*
Power Supply		0.05	**0.5**	%/V	V_{CC} = 12 ±1V. See *Figure 7.*
		0.2		%/V	V_{CC} = ± 5V. See *Figure 7.*
Upper Frequency Limit	**100**	300		kHz	R_0 = 8.2KΩ, C_0 = 400pF
Lowest Practical Operating Frequency			0.01	Hz	R_0 = 2MΩ, C_0 = 50μF
Timing Resistor, R_0 - See *Figure 5*					
Operating Range	**5**		2000	KΩ	
Recommended Range	5			KΩ	See *Figure 7* and *Figure 8.*
Loop Phase Dectector Section					
Peak Output Current	**±150**	±200	**±300**	μA	Measured at Pin 11
Output Offset Current		1		μA	
Output Impedance		1		MΩ	
Maximum Swing	**±4**	± 5		V	Referenced to Pin 10
Quadrature Phase Detector					Measured at Pin 3
Peak Output Current	**100**	300		μA	
Output Impedance		1		MΩ	
Maximum Swing		11		V_{PP}	
Input Preampt Section					Measured at Pin 2
Input Impedance		20		KΩ	
Input Signal					
Voltage Required to Cause Limiting		2	10	mV rms	

Notes

Parameters are guaranteed over the recommended operating conditions, but are not 100% tested in production.

***Bold face parameters** are covered by production test and guaranteed over operating temperature range.*

DC ELECTRICAL CHARACTERISTICS (CONT'D)

Test Conditions: V_{CC} = 12V, T_A = +25°C, R_O = 30KΩ, C_O = 0.033μF, unless otherwise specified.

Parameter	Min.	Typ.	Max.	Unit	Conditions
Voltage Comparator Section					
Input Impedance		2		MΩ	Measured at Pins 3 and 8
Input Bias Current		100		nA	
Voltage Gain	55	70		dB	R_L = 5.1KΩ
Output Voltage Low		300	**500**	mV	I_C = 3mA
Output Leakage Current		0.01	**10**	μA	V_O = 20V
Internal Reference					
Voltage Level	**4.9**	5.3	**5.7**	V	Measured at Pin 10
Output Impedance		100		Ω	AC Small Signal
Maximum Source Current		80		μA	

Notes
Parameters are guaranteed over the recommended operating conditions, but are not 100% tested in production.
***Bold face parameters** are covered by production test and guaranteed over operating temperature range.*

Specifications are subject to change without notice

ABSOLUTE MAXIMUM RATINGS

Power Supply 20V
Input Signal Level ..... 3V rms
Power Dissipation 900mW

Package Power Dissipation Ratings
CDIP 750mW
  Derate Above T_A = 25°C ..... 8mW/°C
PDIP 800mW
  Derate Above T_A = 25°C ..... 60mW/°C
SOIC 390mW
  Derate Above T_A = 25°C ..... 5mW/°C

SYSTEM DESCRIPTION

The main PLL within the XR-2211 is constructed from an input preamplifier, analog multiplier used as a phase detector and a precision voltage controlled oscillator (VCO). The preamplifier is used as a limiter such that input signals above typically 10mV rms are amplified to a constant high level signal. The multiplying-type phase detector acts as a digital exclusive or gate. Its output (unfiltered) produces sum and difference frequencies of the input and the VCO output. The VCO is actually a current controlled oscillator with its normal input current (f_O) set by a resistor (R_0) to ground and its driving current with a resistor (R_1) from the phase detector.

The output of the phase detector produces sum and difference of the input and the VCO frequencies (internally connected). When in lock, these frequencies are f_{IN}+ f_{VCO} (2 times f_{IN} when in lock) and f_{IN} - f_{VCO} (0Hz when lock). By adding a capacitor to the phase detector output, the 2 times f_{IN} component is reduced, leaving a DC voltage that represents the phase difference between the two frequencies. This closes the loop and allows the VCO to track the input frequency.

The FSK comparator is used to determine if the VCO is driven above or below the center frequency (FSK comparator). This will produce both active high and active low outputs to indicate when the main PLL is in lock (quadrature phase detector and lock detector comparator).

PRINCIPLES OF OPERATION

Signal Input (Pin 2): Signal is AC coupled to this terminal. The internal impedance at pin 2 is 20KΩ. Recommended input signal level is in the range of 10mV rms to 3V rms.

Quadrature Phase Detector Output (Pin 3): This is the high impedance output of quadrature phase detector and is internally connected to the input of lock detect voltage comparator. In tone detection applications, pin 3 is connected to ground through a parallel combination of R_D and C_D (see *Figure 3*) to eliminate the chatter at lock detect outputs. If the tone detect section is not used, pin 3 can be left open.

Lock Detect Output, Q (Pin 6): The output at pin 6 is at "low" state when the PLL is out of lock and goes to "high" state when the PLL is locked. It is an open collector type output and requires a pull-up resistor, R_L, to V_{CC} for proper operation. At "low" state, it can sink up to 5mA of load current.

Lock Detect Complement, (Pin 5): The output at pin 5 is the logic complement of the lock detect output at pin 6. This output is also an open collector type stage which can sink 5mA of load current at low or "on" state.

FSK Data Output (Pin 7): This output is an open collector logic stage which requires a pull-up resistor, R_L, to V_{CC} for proper operation. It can sink 5mA of load current. When decoding FSK signals, FSK data output is at "high" or "off" state for low input frequency, and at "low" or "on" state for high input frequency. If no input signal is present, the logic state at pin 7 is indeterminate.

FSK Comparator Input (Pin 8): This is the high impedance input to the FSK voltage comparator. Normally, an FSK post-detection or data filter is connected between this terminal and the PLL phase detector output (pin 11). This data filter is formed by R_F and C_F (see *Figure 3.*) The threshold voltage of the comparator is set by the internal reference voltage, V_{REF}, available at pin 10.

Reference Voltage, V_{REF} (Pin 10): This pin is internally biased at the reference voltage level, V_{REF}: $V_{REF} = V_{CC}/2 - 650mV$. The DC voltage level at this pin forms an internal reference for the voltage levels at pins 5, 8, 11 and 12. Pin 10 must be bypassed to ground with a 0.1μF capacitor for proper operation of the circuit.

Loop Phase Detector Output (Pin 11): This terminal provides a high impedance output for the loop phase detector. The PLL loop filter is formed by R_1 and C_1 connected to pin 11 (see *Figure 3.*) With no input signal, or with no phase error within the PLL, the DC level at pin 11 is very nearly equal to V_{REF}. The peak to peak voltage swing available at the phase detector output is equal to $2 \times V_{REF}$.

VCO Control Input (Pin 12): VCO free-running frequency is determined by external timing resistor, R_0, connected from this terminal to ground. The VCO free-running frequency, f_O, is:

$$f_O = \frac{1}{R_0 \cdot C_0} \; Hz$$

where C_0 is the timing capacitor across pins 13 and 14. For optimum temperature stability, R_0 must be in the range of 10KΩ to 100KΩ (see *Figure 9.*)

This terminal is a low impedance point, and is internally biased at a DC level equal to V_{REF}. The maximum timing current drawn from pin 12 must be limited to ≤ 3mA for proper operation of the circuit.

VCO Timing Capacitor (Pins 13 and 14): VCO frequency is inversely proportional to the external timing capacitor, C_0, connected across these terminals (see *Figure 6.*) C_0 must be non-polar, and in the range of 200pF to 10μF.

VCO Frequency Adjustment: VCO can be fine-tuned by connecting a potentiometer, R_X, in series with R_0 at pin 12 (see *Figure 10.*)

VCO Free-Running Frequency, f_O: XR-2211 does not have a separate VCO output terminal. Instead, the VCO outputs are internally connected to the phase detector sections of the circuit. For set-up or adjustment purposes, the VCO free-running frequency can be tuned by using the generalized circuit in *Figure 3*, and applying an alternating bit pattern of O's and 1's at the known mark and space frequencies. By adjusting R_0, the VCO can then be tuned to obtain a 50% duty cycle on the FSK output (pin 7). This will ensure that the VCO f_O value is accurately referenced to the mark and space frequencies.

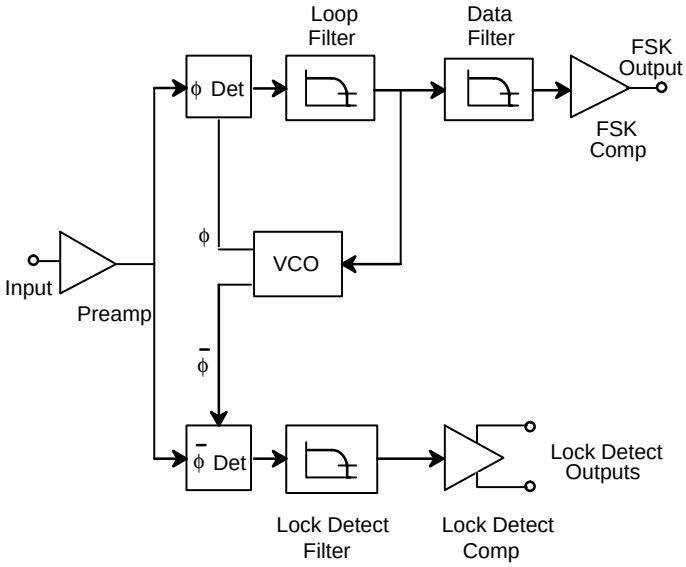


Figure 2. Functional Block Diagram of a Tone and FSK Decoding System Using XR-2211

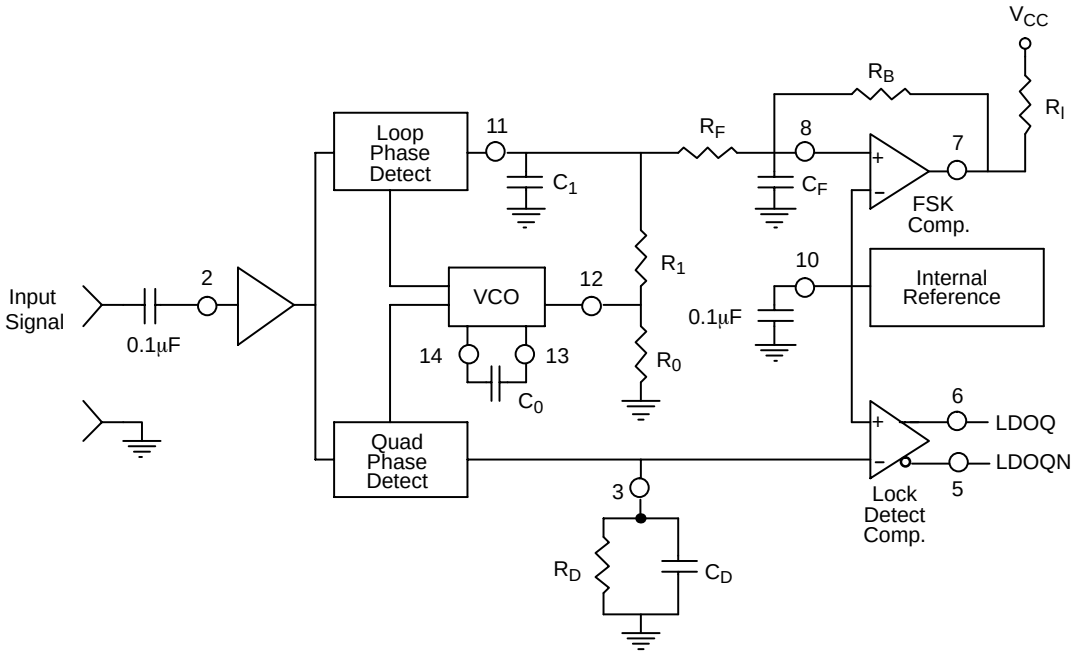


Figure 3. Generalized Circuit Connection for FSK and Tone Detection

DESIGN EQUATIONS

(All resistance in Ω, all frequency in Hz and all capacitance in farads, unless otherwise specified)

(See *Figure 3* for definition of components)

1. VCO Center Frequency, f_O:

$$f_O = \frac{1}{R_0 \cdot C_0}$$

2. Internal Reference Voltage, V_{REF} (measured at pin 10):

$$V_{REF} = \left(\frac{V_{CC}}{2}\right) - 650mV \text{ in volts}$$

3. Loop Low-Pass Filter Time Constant, τ:

$$\tau = C_1 \cdot R_{PP} \text{ (seconds)}$$

where:

$$R_{PP} = \left(\frac{R_1 \cdot R_F}{R_1 + R_F}\right)$$

if R_F is ∞ or C_F reactance is ∞, then R_{PP} = R1

4. Loop Damping, ς:

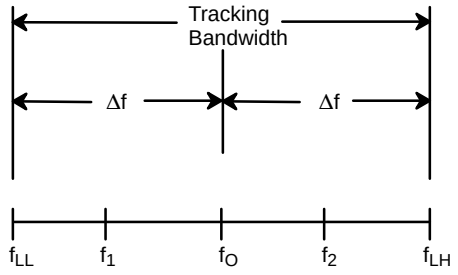
$$\zeta = \sqrt{\left(\frac{1250 \cdot C_0}{R_1 \cdot C_1}\right)}$$

Note: *For derivation/explanation of this equation, please see TAN-011.*

5. Loop-tracking

bandwidth, $\pm = \frac{\Delta f}{f_0}$

$$\frac{\Delta f}{f_0} = \frac{R_0}{R_1}$$



6. FSK Data filter time constant, tF:

$$\tau_F = \frac{R_B \cdot R_F}{(R_B + R_F)} \cdot C_F \ (seconds)$$

7. Loop phase detector conversion gain, Kd: (Kd is the differential DC voltage across pin 10 and pin11, per unit of phase error at phase detector input):

$$K_d = \frac{V_{REF} \cdot R_1}{10,000 \cdot \pi} \left[\frac{volt}{radian}\right]$$

Note: *For derivation/explanation of this equation, please see TAN-011.*

8. VCO conversion gain, Ko: (Ko is the amount of change in VCO frequency, per unit of DC voltage change at pin 11):

$$K_0 = \frac{-2\pi}{V_{REF} \cdot C_0 \cdot R_1} = \left(\frac{radian/second}{volt}\right)$$

9. The filter transfer function:

$$F(s) = \frac{1}{1 + SR_1 \cdot C_1} \ at\ 0\ Hz. \qquad S = J\omega \ and \ \omega = 0$$

10. Total loop gain. K_T:

$$K_T = K_O \cdot K_d \cdot F(s) = \left(\frac{R_F}{5,000 \cdot C_0 \cdot (R_1 + R_F)}\right)\left[\frac{1}{seconds}\right]$$

11. Peak detector current I_A:

$$I_A = \frac{V_{REF}}{20,000} \ (V_{REF} \ in \ volts \ and \ I_A \ in \ amps)$$

Note: *For derivation/explanation of this equation, please see TAN-011.*

APPLICATIONS INFORMATION

FSK Decoding

Figure 10 shows the basic circuit connection for FSK decoding. With reference to *Figure 3* and *Figure 10*, the functions of external components are defined as follows: R_0 and C_0 set the PLL center frequency, R_1 sets the system bandwidth, and C_1 sets the loop filter time constant and the loop damping factor. C_F and R_F form a one-pole post-detection filter for the FSK data output. The resistor R_B from pin 7 to pin 8 introduces positive feedback across the FSK comparator to facilitate rapid transition between output logic states.

Design Instructions:

The circuit of *Figure 10* can be tailored for any FSK decoding application by the choice of five key circuit components: R_0, R_1, C_0, C_1 and C_F. For a given set of FSK mark and space frequencies, f_O and f_1, these parameters can be calculated as follows:

(All resistance in Ω's, all frequency in Hz and all capacitance in farads, unless otherwise specified)

a) Calculate PLL center frequency, f_O:

$$f_O = \sqrt{F_1 \cdot F_2}$$

b) Choose value of timing resistor R_0, to be in the range of 10KΩ to 100KΩ. This choice is arbitrary. The recommended value is R_0 = 20KΩ. The final value of R_0 is normally fine-tuned with the series potentiometer, R_X.

$$R_O = R_O + \frac{R_X}{2}$$

c) Calculate value of C0 from design equation (1) or from *Figure 7*:

$$C_O = \frac{1}{R_0 \cdot f_0}$$

d) Calculate R1 to give the desired tracking bandwidth (See design equation 5).

$$R_1 = \frac{R_0 \cdot f_0}{(f_1 - f_2)} \cdot 2$$

e) Calculate C1 to set loop damping. (See design equation 4):

Normally, ς = 0.5 is recommended.

$$C_1 = \frac{1250 \cdot C_0}{R_1 \cdot \varsigma^2}$$

f) The input to the XR-2211 may sometimes be too sensitive to noise conditions on the input line. *Figure 4* illustrates a method of de-sensitizing the XR-2211 from such noisy line conditions by the use of a resistor, Rx, connected from pin 2 to ground. The value of Rx is chosen by the equation and the desired minimum signal threshold level.

$$V_{IN}\ minimum\ (peak) = V_a - V_b = \Delta V \pm 2.8mV\ offset = V_{REF}\frac{20,000}{(20,000 + R_X)}\ or\ R_X = 20,000\left(\frac{V_{REF}}{\Delta V} - 1\right)$$

V_{IN} minimum (peak) input voltage must exceed this value to be detected (equivalent to adjusting V threshold)

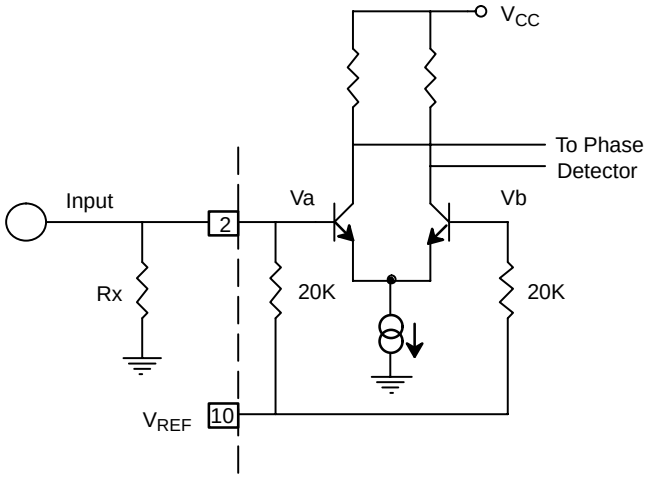


Figure 4. Desensitizing Input Stage

g) Calculate Data Filter Capacitance, C_F:

$$R_{sum} = \frac{(R_F + R_1)\cdot R_B}{(R_1 + R_F + R_B)}$$

$$C_F = \frac{0.25}{(R_{sum}\cdot Baud\ Rate)} \qquad Baud\ rate\ in\ \frac{1}{seconds}$$

Note: *All values except R_0 can be rounded to nearest standard value.*

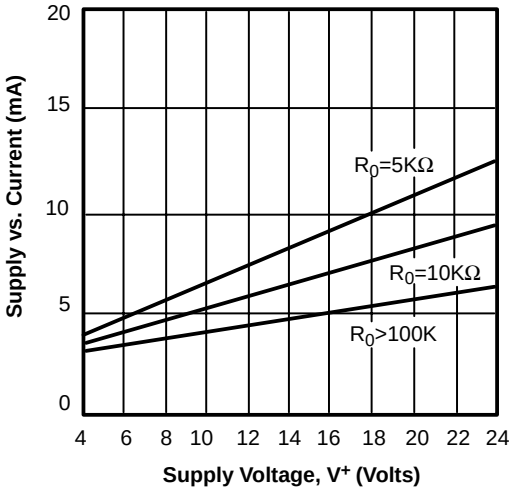


Figure 5. Typical Supply Current vs. V+ (Logic Outputs Open Circuited)

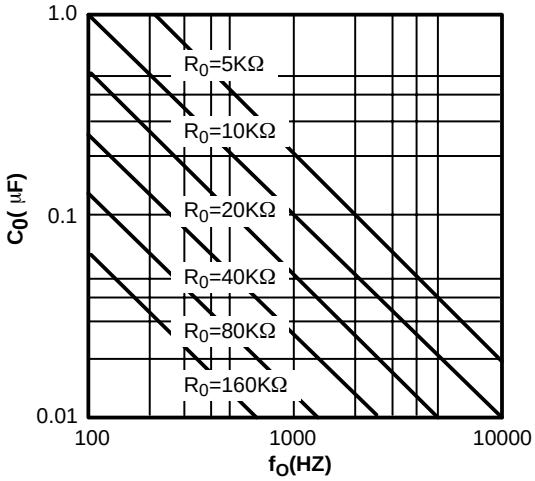


Figure 6. VCO Frequency vs. Timing Resistor

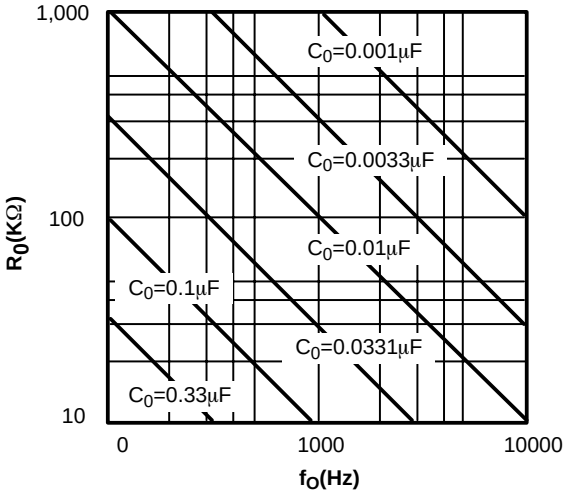


Figure 7. VCO Frequency vs. Timing Capacitor

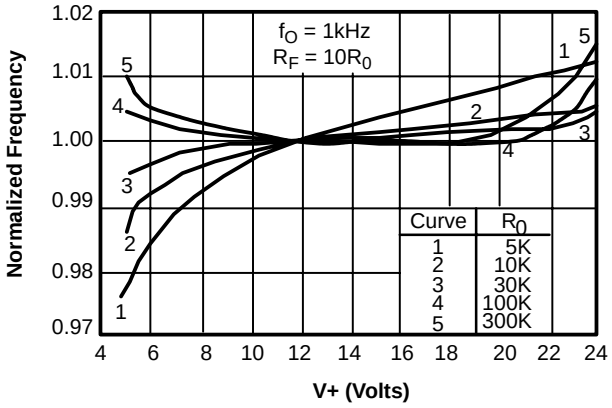


Figure 8. Typical f_O vs. Power Supply Characteristics

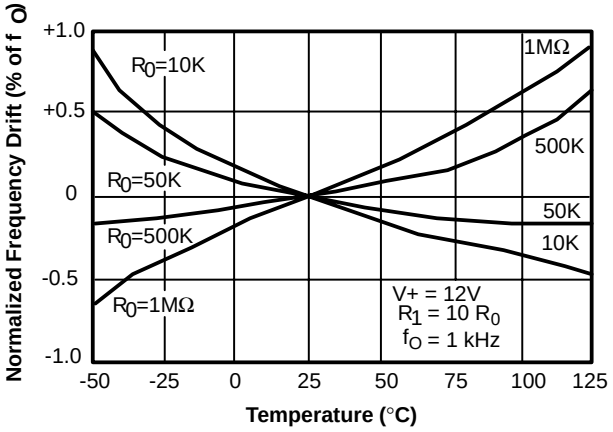


Figure 9. Typical Center Frequency Drift vs. Temperature

Design Example:

1200 Baud FSK demodulator with mark and space frequencies of 1200/2200.

Step 1: Calculate f_O: from design instructions

(*a*) $f_O = \sqrt{1200 \cdot 2200} = 1624$

Step 2: Calculate R_0 : R_0 =10K with a potentiometer of 10K. (See design instructions (b))

(*b*) $R_T = 10 + \left(\frac{10}{2}\right) = 15K$

Step 3: Calculate C_0 from design instructions

(*c*) $C_O = \frac{1}{15000 \cdot 1624} = 39nF$

Step 4: Calculate R_1 : from design instructions

(*d*) $R_1 = \frac{20000 \cdot 1624 \cdot 2}{(2200 - 1200)} = 51,000$

Step 5: Calculate C_1 : from design instructions

(*e*) $C_1 = \frac{1250 \cdot 39nF}{51000 \cdot 0.5^2} = 3.9nF$

Step 6: Calculate R_F : R_F should be at least five times R_1, R_F = 51,000·5 = 255 KΩ

Step 7: Calculate R_B : R_B should be at least five times R_F, R_B = 255,000·5 = 1.2 MΩ

Step 8: Calculate R_{SUM} :

$$R_{SUM} = \frac{(R_F + R_1) \cdot R_B}{(R_F + R_1 + R_B)} = 240K\Omega$$

Step 9: Calculate C_F :

$$C_F = \frac{0.25}{(R_{SUM} \cdot Baud\ Rate)} = 1nF$$

Note: *All values except R_0 can be rounded to nearest standard value.*

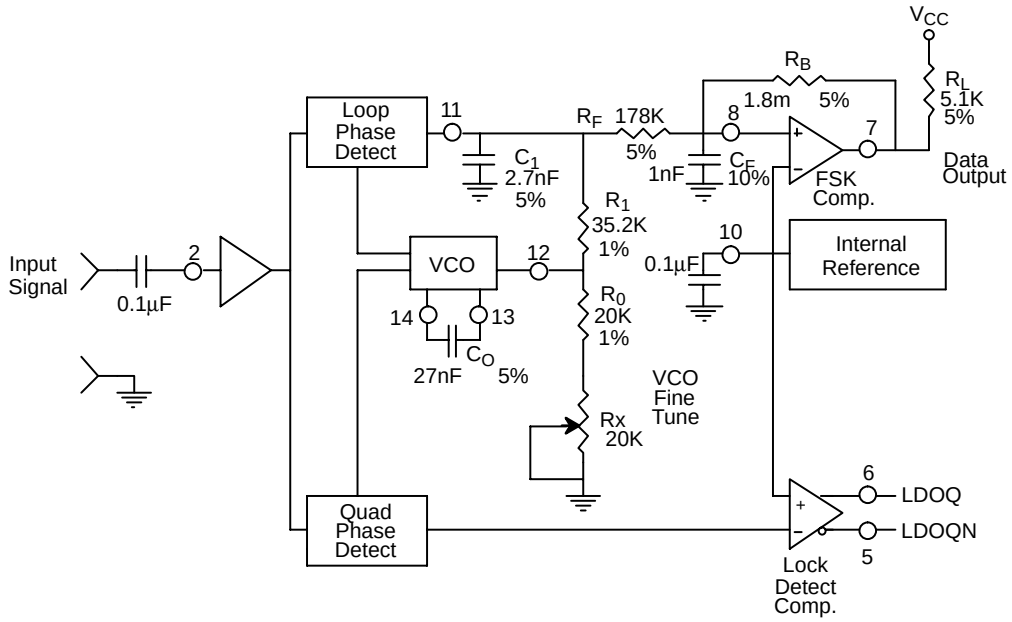


Figure 10. Circuit Connection for FSK Decoding of Caller Identification Signals (Bell 202 Format)

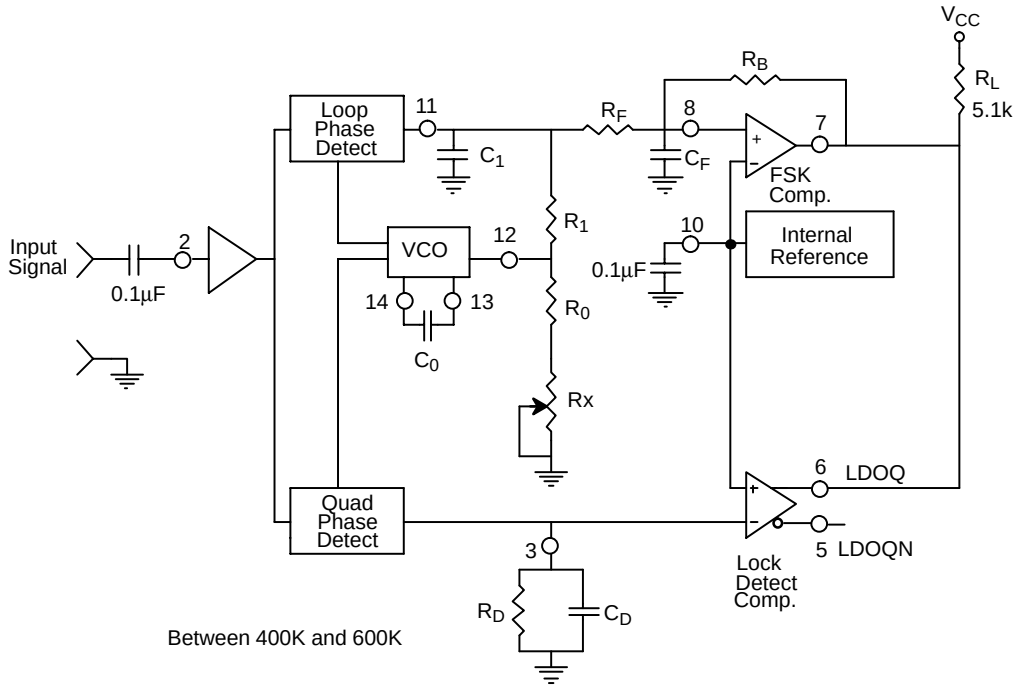


Figure 11. External Connectors for FSK Demodulation with Carrier Detect Capability

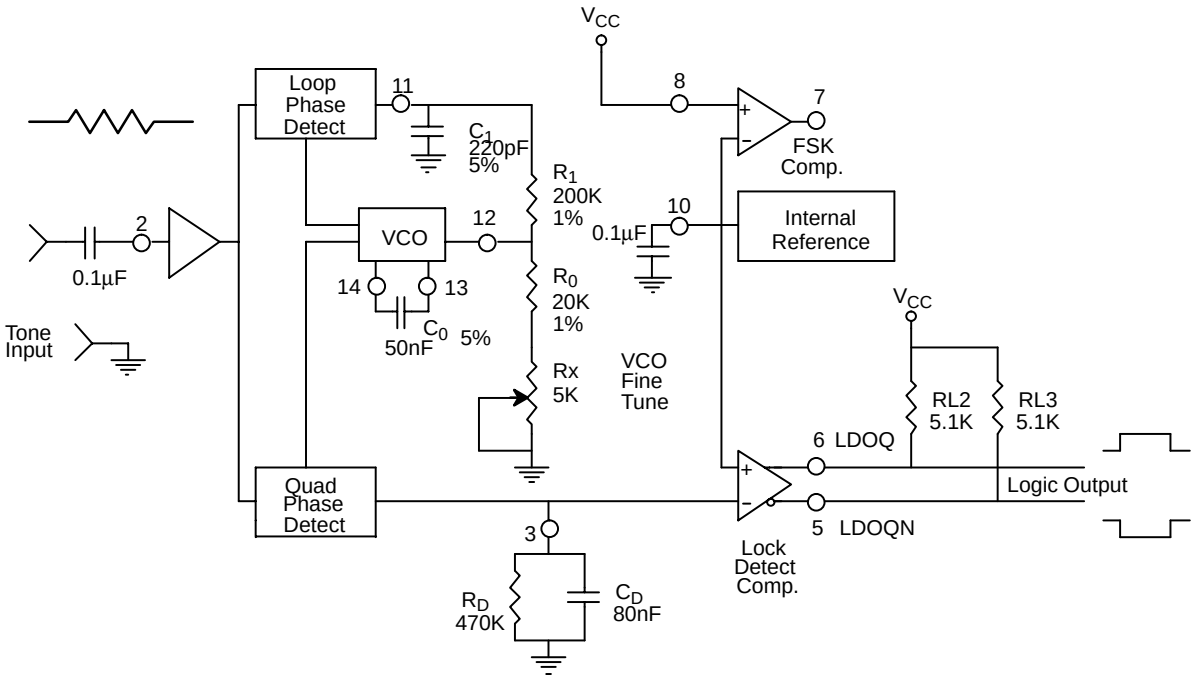


Figure 12. Circuit Connection for Tone Detection

FSK Decoding with Carrier Detect

The lock detect section of XR-2211 can be used as a carrier detect option for FSK decoding. The recommended circuit connection for this application is shown in *Figure 11.* The open collector lock detect output, pin 6, is shorted to data output (pin 7). Thus, data output will be disabled at "low" state, until there is a carrier within the detection band of the PLL and the pin 6 output goes "high" to enable the data output.

Note: *Data Output is "Low" When No Carrier is Present.*

The minimum value of the lock detect filter capacitance C_D is inversely proportional to the capture range, $\pm\Delta fc$. This is the range of incoming frequencies over which the loop can acquire lock and is always less than the tracking range. It is further limited by C_1. For most applications, $\Delta fc > \Delta f/2$. For $R_D = 470K\Omega$, the approximate minimum value of C_D can be determined by:

$$C_D > \frac{16}{\Delta f} \quad C \text{ in } \mu F \text{ and } f \text{ in Hz.}$$

C in µF and f in Hz.

With values of C_D that are too small, chatter can be observed on the lock detect output as an incoming signal frequency approaches the capture bandwidth. Excessively large values of C_D will slow the response time of the lock detect output. For Caller I.D. applications choose $C_D = 0.1\mu F$.

Tone Detection

Figure 12 shows the generalized circuit connection for tone detection. The logic outputs, LDOQN and LDOQ at pins 5 and 6 are normally at "high" and "low" logic states, respectively. When a tone is present within the detection band of the PLL, the logic state at these outputs become reversed for the duration of the input tone. Each logic output can sink 5mA of load current.

Both outputs at pins 5 and 6 are open collector type stages, and require external pull-up resistors R_{L2} and R_{L3}, as shown in *Figure 12.*

With reference to *Figure 3* and *Figure 12*, the functions of the external circuit components can be explained as follows: R_0 and C_0 set VCO center frequency; R_1 sets the detection bandwidth; C_1 sets the low pass-loop filter time constant and the loop damping factor.

Design Instructions:

The circuit of *Figure 12* can be optimized for any tone detection application by the choice of the 5 key circuit components: R_0, R_1, C_0, C_1 and C_D. For a given input, the tone frequency, f_S, these parameters are calculated as follows:

(All resistance in Ω's, all frequency in Hz and all capacitance in farads, unless otherwise specified)

a) Choose value of timing resistor R_0 to be in the range of 10KΩ to 50KΩ. This choice is dictated by the max./min. current that the internal voltage reference can deliver. The recommended value is R_0 = 20KΩ. The final value of R_0 is normally fine-tuned with the series potentiometer, R_X.

b) Calculate value of C_0 from design equation (1) or from *Figure 7* $f_S = f_O$:

$$C_O = \frac{1}{R_0 \cdot fs}$$

c) Calculate R_1 to set the bandwidth $\pm\Delta f$ (See design equation 5):

$$R_1 = \frac{R_0 \cdot f_0 \cdot 2}{\Delta f}$$

Note: *The total detection bandwidth covers the frequency range of* $f_O \pm \Delta f$

d) Calculate value of C_1 for a given loop damping factor:

Normally, $\varsigma = 0.5$ is recommended.

$$C_1 = \frac{1250 \cdot C_0}{R_1 \cdot \varsigma^2}$$

Increasing C_1 improves the out-of-band signal rejection, but increases the PLL capture time.

e) Calculate value of the filter capacitor C_D . To avoid chatter at the logic output, with R_D = 470KΩ, C_D must be:

$$C_D > \frac{16}{\Delta f} \quad C \text{ in } \mu F$$

Increasing C_D slows down the logic output response time.

Design Examples:

Tone detector with a detection band of ± 100Hz:

a) Choose value of timing resistor R_0 to be in the range of 10KΩ to 50KΩ. This choice is dictated by the max./min. current that the internal voltage reference can deliver. The recommended value is R_0 = 20 KΩ. The final value of R_0 is normally fine-tuned with the series potentiometer, R_X.

b) Calculate value of C_0 from design equation (1) or from *Figure 6* $f_S = f_O$:

$$C_0 = \frac{1}{R_0 \cdot f_S} = \frac{1}{20,000 \cdot 1,000} = 50nF$$

c) Calculate R_1 to set the bandwidth $\pm\Delta f$ (See design equation 5):

$$R_1 = \frac{R_0 \cdot f_O \cdot 2}{\Delta f} = \frac{20,000 \cdot 1,000 \cdot 2}{100} = 400K$$

Note: *The total detection bandwidth covers the frequency range of $f_O \pm \Delta f$*

d) Calculate value of C_0 for a given loop damping factor:

Normally, $\varsigma = 0.5$ is recommended.

$$C_1 = \frac{1250 \cdot C_0}{R_1 \cdot \varsigma^2} = \frac{1250 \cdot 50 \cdot 10^{-9}}{400,000 \cdot 0.5^2} = 6.25pF$$

Increasing C_1 improves the out-of-band signal rejection, but increases the PLL capture time.

e) Calculate value of the filter capacitor C_D . To avoid chatter at the logic output, with R_D = 470KΩ, C_D must be:

$$C_D = \frac{16}{\Delta f} \geq \frac{16}{200} \geq 80nF$$

Increasing C_D slows down the logic output response time.

f) Fine tune center frequency with 5KΩ potentiometer, R_X.

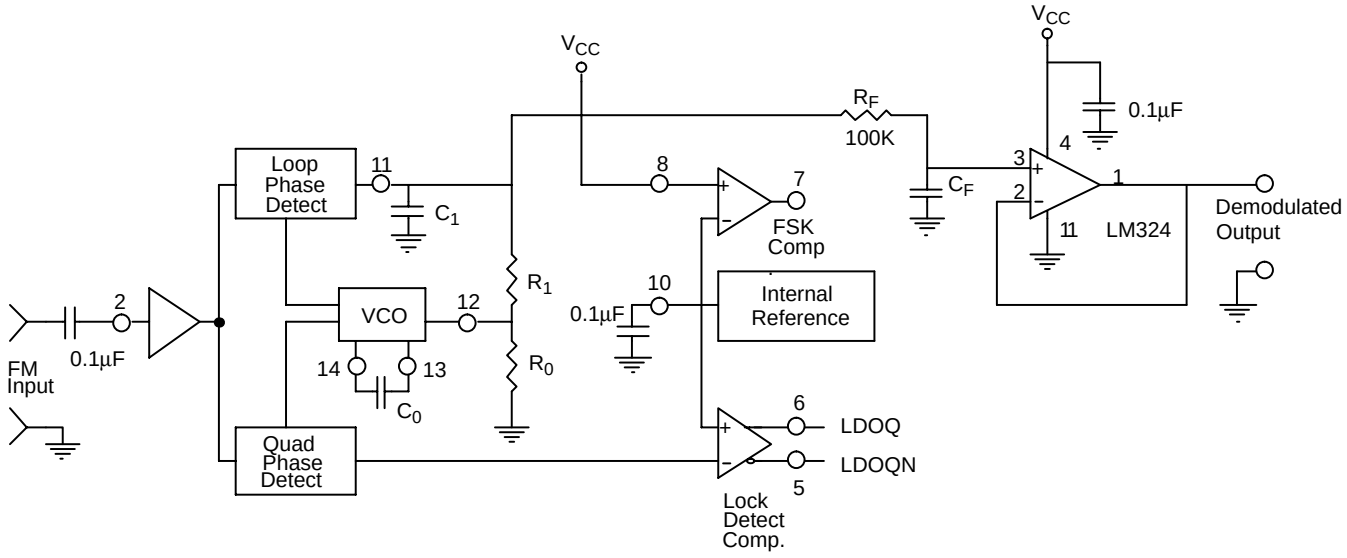


Figure 13. Linear FM Detector Using XR-2211 and an External Op Amp.
(See Section on Design Equation for Component Values.)

Linear FM Detection

XR-2211 can be used as a linear FM detector for a wide range of analog communications and telemetry applications. The recommended circuit connection for this application is shown in *Figure 13.* The demodulated output is taken from the loop phase detector output (pin 11), through a post-detection filter made up of R_F and C_F, and an external buffer amplifier. This buffer amplifier is necessary because of the high impedance output at pin 11. Normally, a non-inverting unity gain op amp can be used as a buffer amplifier, as shown in *Figure 13.*

The FM detector gain, i.e., the output voltage change per unit of FM deviation can be given as:

$$V_{OUT} = \frac{R_1 \cdot V_{REF}}{100 \cdot R_0}$$

where VR is the internal reference voltage ($V_{REF} = V_{CC}/2$ - 650mV). For the choice of external components R_1, R_0, C_D, C_1 and C_F, see the section on design equations.

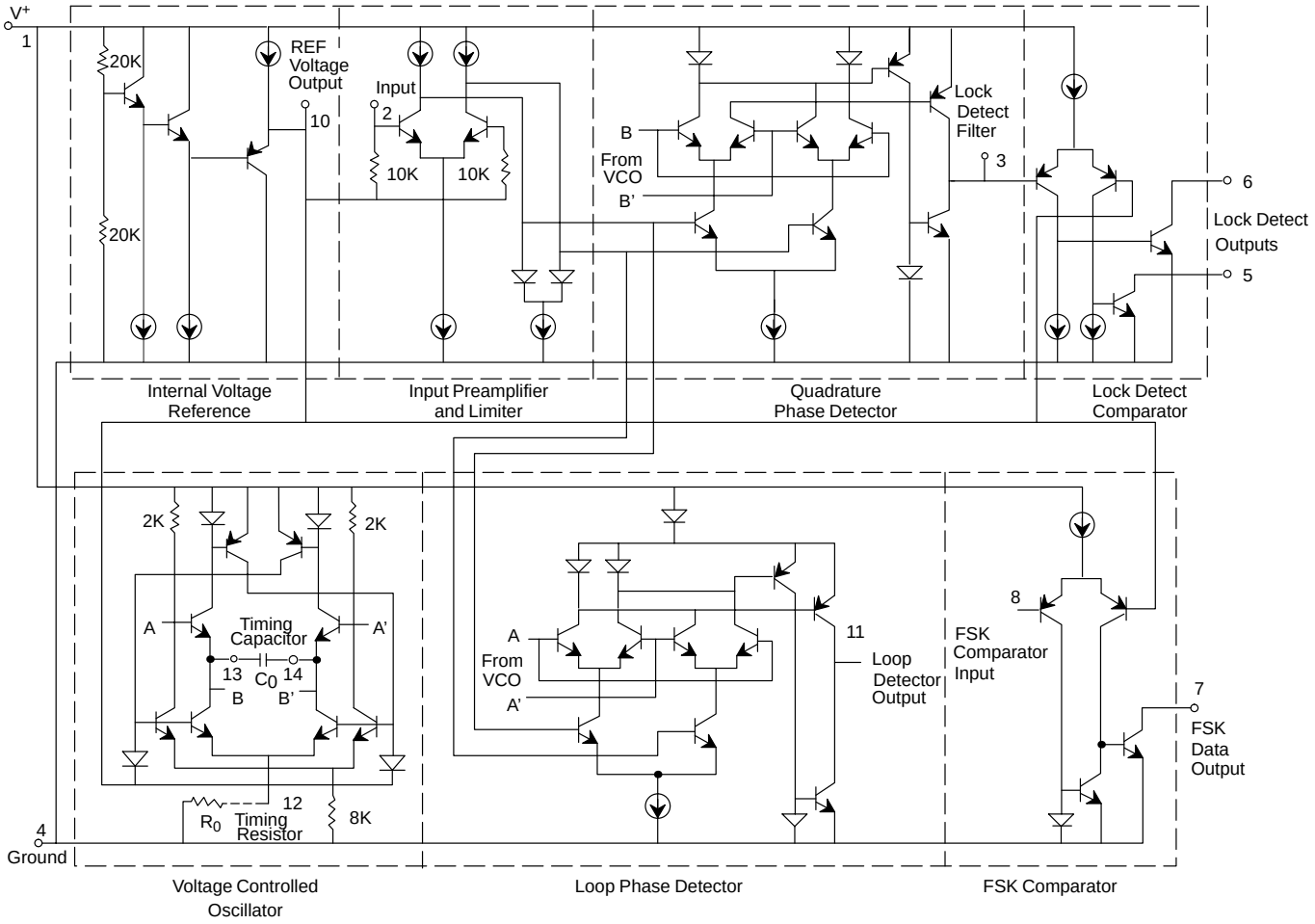
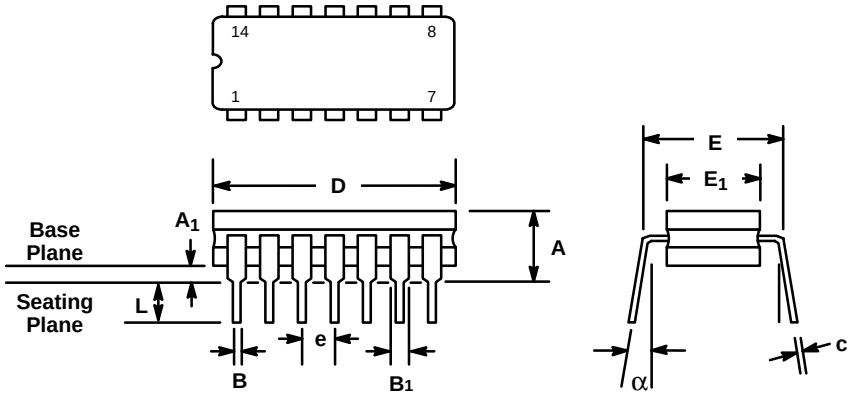


Figure 14. Equivalent Schematic Diagram

14 LEAD CERAMIC DUAL-IN-LINE (300 MIL CDIP)

Rev. 1.00

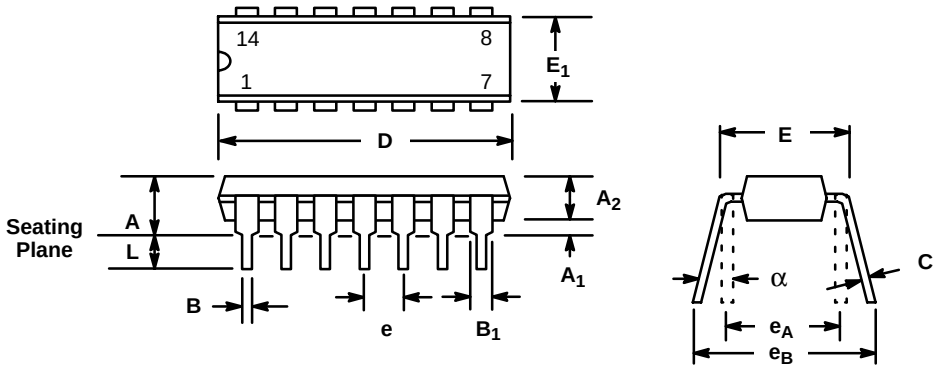


SYMBOL	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.100	0.200	2.54	5.08
A_1	0.015	0.060	0.38	1.52
B	0.014	0.026	0.36	0.66
B_1	0.045	0.065	1.14	1.65
c	0.008	0.018	0.20	0.46
D	0.685	0.785	17.40	19.94
E_1	0.250	0.310	6.35	7.87
E	0.300 BSC		7.62 BSC	
e	0.100 BSC		2.54 BSC	
L	0.125	0.200	3.18	5.08
α	0°	15°	0°	15°

Note: The control dimension is the inch column

14 LEAD PLASTIC DUAL-IN-LINE (300 MIL PDIP)

Rev. 1.00

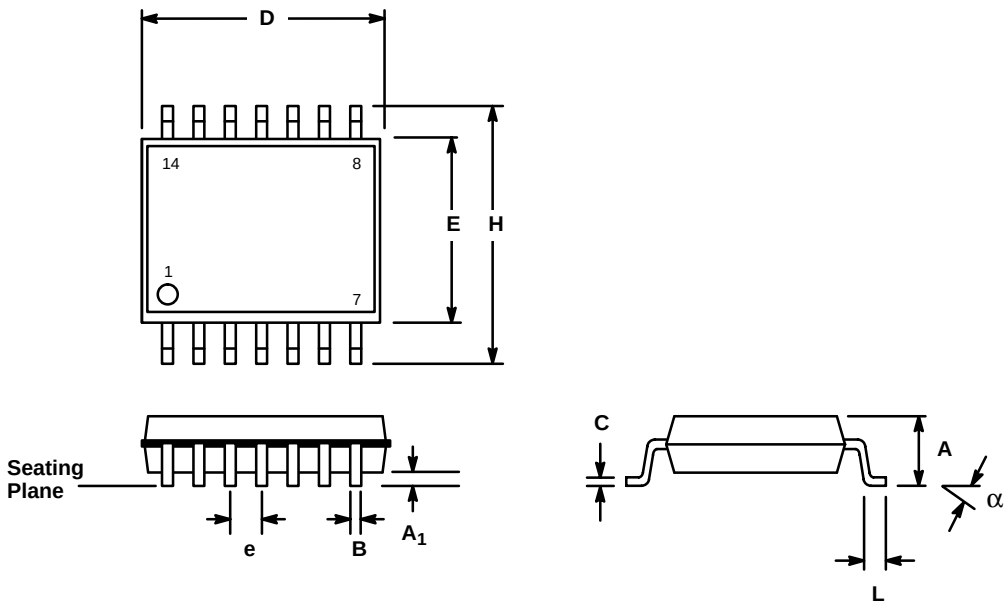


SYMBOL	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.145	0.210	3.68	5.33
A_1	0.015	0.070	0.38	1.78
A_2	0.115	0.195	2.92	4.95
B	0.014	0.024	0.36	0.56
B_1	0.030	0.070	0.76	1.78
C	0.008	0.014	0.20	0.38
D	0.725	0.795	18.42	20.19
E	0.300	0.325	7.62	8.26
E_1	0.240	0.280	6.10	7.11
e	0.100 BSC		2.54 BSC	
e_A	0.300 BSC		7.62 BSC	
e_B	0.310	0.430	7.87	10.92
L	0.115	0.160	2.92	4.06
α	0°	15°	0°	15°

Note: The control dimension is the inch column

14 LEAD SMALL OUTLINE
(150 MIL JEDEC SOIC)

Rev. 1.00



SYMBOL	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.053	0.069	1.35	1.75
A_1	0.004	0.010	0.10	0.25
B	0.013	0.020	0.33	0.51
C	0.007	0.010	0.19	0.25
D	0.337	0.344	8.55	8.75
E	0.150	0.157	3.80	4.00
e	0.050 BSC		1.27 BSC	
H	0.228	0.244	5.80	6.20
L	0.016	0.050	0.40	1.27
α	0°	8°	0°	8°

Note: The control dimension is the millimeter column

Notes

Notes

NOTICE

EXAR Corporation reserves the right to make changes to the products contained in this publication in order to improve design, performance or reliability. EXAR Corporation assumes no responsibility for the use of any circuits described herein, conveys no license under any patent or other right, and makes no representation that the circuits are free of patent infringement. Charts and schedules contained here in are only for illustration purposes and may vary depending upon a user's specific application. While the information in this publication has been carefully checked; no responsibility, however, is assumed for inaccuracies.

EXAR Corporation does not recommend the use of any of its products in life support applications where the failure or malfunction of the product can reasonably be expected to cause failure of the life support system or to significantly affect its safety or effectiveness. Products are not authorized for use in such applications unless EXAR Corporation receives, in writing, assurances to its satisfaction that: (a) the risk of injury or damage has been minimized; (b) the user assumes all such risks; (c) potential liability of EXAR Corporation is adequately protected under the circumstances.

Copyright 1995 EXAR Corporation
Datasheet June 1997
Reproduction, in part or whole, without the prior written consent of EXAR Corporation is prohibited.

This datasheet has been download from:

www.datasheetcatalog.com

Datasheets for electronics components.

Anexo D

Fuente de Poder Conmutada