

T  
629.895  
TER



# Escuela Superior Politécnica del Litoral

## Facultad de Ingeniería en Electricidad y Computación

“Monitoreo de Líneas de Cargadores de Baterías.  
Diseño e Implementación de la Interfase Serial  
RS-232 entre los Microcontroladores PIC y la PC”

### TESIS DE GRADO



Previa a la obtención del Título de:  
**Ingeniero en Electricidad**

Especialización

**En Electrónica y Automatización Industrial**

Presentada por:

**Andrés Fernando Terán Cedeño**

GUAYAQUIL - ECUADOR

**Año: 2006**



# AGRADECIMIENTO

A la empresa privada que permitieron el estudio y desarrollo de este trabajo en sus instalaciones y procesos, tal es el caso de Baterías Lux e Industrias Dacar.



CIB-ESPOL

# DEDICATORIA

A MI QUERIDA MADRE

A MI HERMANA

# TRIBUNAL DE GRADUACION



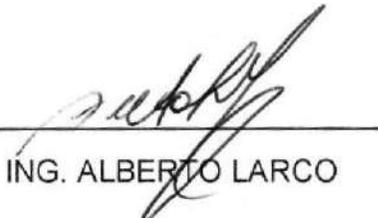
---

ING. MIGUEL YAPUR  
SUBDECANO DE LA FIEC



---

ING. WILMER NARANJO  
DIRECTOR DE TESIS



---

ING. ALBERTO LARCO  
VOCAL



---

ING. HOLGER CEVALLOS  
VOCAL

# DECLARACION EXPRESA

“La responsabilidad del contenido de esta Tesis de Grado, me corresponden exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITECNICA DEL LITORAL”



CIB-ESPOL

A handwritten signature in black ink, which appears to read "Andres Teran Cedeño", is written over a horizontal line.

ANDRES TERAN CEDEÑO

# RESUMEN

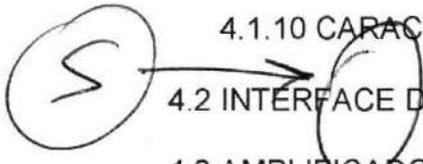
El desarrollo de este proyecto diseña e implementa un sistema que monitorea la carga o formación de líneas de baterías. Este sistema consta de hardware que está formado por dos módulos: un módulo que se encuentra en cada cargador de baterías y que recoge los datos de carga periódicamente, los cuales son voltaje, corriente y temperatura (parámetros importantes en la formación de una batería), llamado módulo Esclavo y el otro módulo, llamado Maestro, es el que realiza la comunicación con los módulos esclavos y luego que obtiene los datos de todos, los organiza en un formato definido en el sistema y los envía a la PC para su posterior análisis. La otra parte del sistema es un software realizado en Visual Basic, que con los datos que recoge el hardware se puede tener una visualización de las líneas de carga de baterías en 3D con varios colores de acuerdo a los diferentes estados de la carga y muestra los parámetros de voltaje, corriente, temperatura, amperios-horas y horas de carga de cada línea, realiza curvas de carga, al final y durante el proceso de carga o de procesos anteriores, consulta en tablas los procesos de carga de acuerdo al tipo de batería y línea en la que se formó, cuadro estadístico de número de baterías formadas por mes de acuerdo al tipo de batería requerido, tabla de alarmas, archivos históricos de carga en una base de datos.

# INDICE GENERAL

	Pág.
RESUMEN	VI
INDICE GENERAL	VII
INDICE DE FIGURAS	XII
INDICE DE TABLAS	XXII
INTRODUCCION	1
<b>1.- INTRODUCCION A LOS SISTEMAS DE MONITOREO</b>	<b>3</b>
1.1 UTILIDAD DE LOS SISTEMAS DE MONITOREO.	3
1.2 OBJETIVOS.	6
1.3 ESTUDIO Y DESARROLLO.	6
1.4 CONTRIBUCION DEL SISTEMA DE MONITOREO A LA	
FABRICACION DE BATERIAS.	11
<b>2.- INTRODUCCION A LAS BATERIAS.</b>	<b>14</b>
2.1 REVISION DE LA FABRICACION DE BATERIAS.	14
2.2 FORMACION Y CARGA DE BATERIAS.	29
2.2.1 EQUIPO REQUERIDO	30



2.2.2 INSTRUCCIONES DE CARGA.	34
2.2.3 TIPOS DE CARGA.	40
2.2.4 PRUEBA DE ALTA INTENSIDAD DE DESCARGA DE LA BATERIA.	57
<b>3.- SOFTWARE "SISTEMA DE MONITOREO".</b>	<b>62</b>
3.1 DESCRIPCION DEL SOFTWARE.	62
3.2 PANTALLAS DEL PROGRAMA Y SU FUNCIONAMIENTO.	66
3.2.1 PANTALLAS PRINCIPALES.	67
3.2.2 PANTALLAS DE TABLAS.	83
3.2.3 PANTALLAS DE CONFIGURACION.	105
3.3 CARACTERISTICAS ESPECIALES	117
3.3.1 PUERTO SERIAL.	118
3.3.2 BASES DE DATOS.	119
3.3.3 ARCHIVOS ANUALES.	130
<b>4.- DESCRIPCION DE LOS ELEMENTOS DEL SISTEMA.</b>	<b>132</b>
4.1 MICROCONTROLADOR PIC16F877A	132
4.1.1 EL MICROCONTROLADOR.	133
4.1.2 PIC16F877A Y SUS PRESTACIONES.	134

4.1.3 ARQUITECTURA INTERNA.	141
4.1.4 ORGANIZACION DE LA MEMORIA Y REGISTROS DE CONTROL.	147
4.1.5 PUERTAS DE ENTRADA/SALIDA.	182
4.1.6 CONVERTIDOR ANALOGICO-DIGITAL.	199
4.1.7 TEMPORIZADORES.	210
4.1.8 COMUNICACION SERIAL I2C.	218
4.1.9 COMUNICACION USART.	240
4.1.10 CARACTERISTICAS ESPECIALES.	254
 4.2 INTERFACE DE COMUNICACION MAXIM-RS232A.	262
4.3 AMPLIFICADOR OPERACIONAL LM324.	265
4.4 DIODO ZENER LM336.	270
4.5 OPTOAISLADOR 4N35.	274
4.6 SENSORES	276
4.6.1 TEMPERATURA.	277
4.6.2 CORRIENTE.	284
4.6.3 VOLTAJE.	285
<b>5.- DISEÑO DEL HARDWARE Y FIRMWARE.</b>	<b>287</b>

5.1 INTERCONEXION DEL SISTEMA Y FUNCIONAMIENTO DEL FIRMWARE.	288
5.2 MODULO MAESTRO.	293
5.2.1 MODULOS DE COMUNICACION I2C.	294
5.2.2 MODULO DE COMUNICACION USART.	326
5.2.3 MODULO DE CONTROL DE MEMORIA EEPROM	339
5.3 MODULO ESCLAVO.	342
5.3.1 MODULO DE AMPLIFICACION Y ACONDICIONADOR DE SEÑALES.	347
5.3.2 MODULO DE CONVERSION ANALOGICO-DIGITAL.	370
5.3.3 MODULO DE MUESTREO Y PROMEDIO DE SEÑALES ANALOGICAS.	382
5.3.4 MODULOS ESPECIALES.	388
 <b>CONCLUSIONES Y RECOMENDACIONES.</b>	
 <b>ANEXOS</b>	
 A. ESPECIFICACIONES ELECTRICAS DE LOS ELEMENTOS DEL SISTEMA.	

- B. DIAGRAMAS, ESQUEMATICOS DE TARJETAS ESCLAVO Y MAESTRO.
- C. DIAGRAMAS DE CONEXIONES DE LOS MODULOS MAESTRO-ESCLAVOS Y CIRCUITOS IMPRESOS.
- D. CODIGO FUENTE DE LOS MODULOS ESCLAVO, MAESTRO Y PUESTA EN MARCHA.
- E. MONTAJE Y COSTOS DEL SISTEMA DE MONITOREO.

**BIBLIOGRAFIA.**



CIB-ESPOL

## INDICE DE FIGURAS

Figura 3.1	Portada de presentación del software Sistema de Monitoreo.....	63
Figura 3.2	Barra del Menú del programa.....	66
Figura 3.3	Pantalla del Monitoreo de Baterías.....	69
Figura 3.4	Pantalla de Gráficos, grafica un ciclo de carga Completa.....	73
Figura 3.5	Variables a graficar en la Barra de estado.....	74
Figura 3.6	Forma de presentación de los Gráficos. En a) se encuentra graficado en 2 dimensiones y En b) en 3 dimensiones.....	78
Figura 3.7	Pantalla de Estadístico.....	80
Figura 3.8	Gráfico del estadístico de baterías cargadas por mes.....	83
Figura 3.9	Tabla de Datos.....	85
Figura 3.10	a) Cuadro para seleccionar el tipo de batería, b) Cuadro de Selección BOSCH, pide el número de línea, c) Cuadro de Selección BOSCH, Pide el año del archivo.....	87

Figura 3.11	Tabla de archivos, son todos los archivos de Carga de un Cierta tipo de batería.....	88
Figura 3.12	Tabla Programación y régimen de Carga.....	90
Figura 3.13	Cuadro de Calibraciones de las variables monitoreadas...	94
Figura 3.14	Tabla de Calibraciones donde se pueden observar Los datos calibrados, la primera columna indica el Número de línea, la segunda los milivoltios en el Shunt, la tercera la corriente medida en un instrumento de medición, la cuarta y quinta temperatura y Voltaje medidos.....	101
Figura 3.15	Botón de Alarma.....	102
Figura 3.16	Tabla de Alarmas.....	103
Figura 3.17	Pantalla de Configuración, etiqueta de Comunicaciones.....	107
Figura 3.18	Configuración, Etiqueta Dispositivos, contiene una tabla con Dispositivos, Ubicación del Dispositivo, Datos por Dispositivo, Desde Línea.....	111
Figura3.19	Configuración, etiqueta Especificaciones, las especificaciones eléctricas y de mantenimiento se encuentran en el cuadro central.....	113

Figura 3.20	Errores, etiqueta Estado de Transmisión, el estado de la transmisión de datos por el puerto de la PC se observa en el cuadro central.....	115
Figura 3.21	Configuración, etiqueta Contraseñas, las contraseñas son Inicio de Sesión, Cerrar Sesión y Contraseña de Configuración.....	117
Figura 3.22	Base de Datos Reg.Carga, sirve para organizar datos del sistema.....	122
Figura3.23	Tabla Calibración de la base de datos Reg.Carga.....	123
Figura3.24	Tabla Configuraciones, almacena valores del sistema y Contraseñas.....	124
Figura 3.25	Tabla OrdenLineas.....	125
Figura 3.26	Tabla Régimen_Carga.....	126
Figura3.27	Base de datos Archivo2003, esta base se crea Automáticamente al instalar el programa.....	127
Figura3.28	Tabla Estadístico.....	129
Figura 3.29	Tabla Línea1.....	130
Figura 3.30	Carpeta Sistema de Monitoreo con las bases de datos.....	131
Figura 4.1	Pic16F877A, es la serie de los Pics en este proyecto.....	134

Figura 4.2	Diferencias entre Arquitectura Harvard y Von-Neuman.....	143
Figura 4.3	Arquitectura de los Pic16F877A de 40 pines.....	147
Figura 4.4	Organización de la memoria Tipo Flash.....	149
Figura 4.5	Distribución de la memoria RAM.....	151
Figura 4.6.	Direccionamiento directo e indirecto para acceder a la memoria de datos.....	154
Figura 4.7	Se muestra un ciclo Tcy, que consta de 4 ciclos.....	159
Figura 4.8	Unidad lógica Aritmética.....	161
Figura 4.9	Diagrama de Conexiones de las fuentes de Interrupciones.....	171
Figura 4.10	Esquema de Conexiones de la Puerta A.....	185
Figura 4.11	Esquema de Conexiones de la Puerta A.....	186
Figura 4.12.	Diagrama de Conexiones de la Puerta B.....	190
Figura 4.13.	Diagrama de Conexiones de la Puerta B.....	191
Figura 4.14	Diagrama de Conexiones de la Puerta C.....	193
Figura 4.15	Diagrama de Conexiones de la Puerta C.....	194
Figura 4.16	Diagrama de Conexiones de la Puerta D.....	197
Figura 4.17	Diagrama de Conexiones de la Puerta E.....	199
Figura 4.18	Justificación del resultado de la conversión en ADRESH:	

	ADRESL.....	204
Figura 4.19	Diagrama de la estructura interna del conversor A/D.....	205
Figura 4.20	Diagrama de la entrada al conversor A/D.....	206
Figura 4.21	Diagrama de Conexiones del TMR0.....	212
Figura.4.22	Diagrama de Bloques del TMR1.....	216
Figura 4.23	Diagrama de Bloques del TMR2.....	218
Figura 4.24	EL Pic maestro se conecta a todos los esclavos en el bus I2C con solo dos líneas SDA de datos Y SCL los impulsos del reloj.....	219
Figura 4.25	Conexiones al bus.....	222
Figura 4.26	Diagrama de bloques del módulo I2C modo esclavo.....	228
Figura 4.27	Estructura interna del módulo I2C modo maestro.....	236
Figura 4.28	Esquema de la comunicación USART.....	248
Figura 4.29	Diagrama del módulo USART como transmisor.....	250
Figura 4.30	Diagrama de conexiones del USART como Receptor.....	252
Figura 4.31	Diagrama de Conexiones para la generación de un Reset.....	256
Figura 4.32	Conexión Externa para generar un Reset.....	257
Figura 4.33	Diagrama de tiempos de las señales que provocan un	

	Reset.....	258
Figura 4.34	Diagrama de Bloques del WDT.....	261
Figura 4.35	Estructura interna del Maxim-RS-232A.....	265
Figura 4.36	Muestra la asignación de pines del LM324.....	270
Figura 4.37	Muestra la configuración para ajustar el voltaje de ruptura coeficiente de Temperatura del LM336.....	272
Figura 4.38	Muestra la asignación de pines del LM336.....	274
Figura 4.39	Muestra la configuración interna del optoaislador 4N35, Consta de un diodo emisor y un fototransistor.....	276
Figura 4.40	Sensor de Temperatura LM334.....	277
Figura 4.41	Diagrama de conexiones.....	279
Figura 4.42	Típica conexión para mediciones remotas de temperatura con dos hilos de cable.....	284
Figura 4.43	Divisor de Voltaje usado para medir el valor del Voltaje de las Baterías.....	286
Figura 5.1	Interconexión del sistema.....	289
Figura 5.2	Diagrama de Bloques de la tarjeta maestro.....	292
Figura 5.3	Diagrama de Bloques de las tarjetas esclavo.....	292
Figura 5.4	Conexión de varios dispositivos al bus I2C.....	303
Figura 5.5	Validez del Dato.....	304

Figura 5.6	Generación de la condición de Inicio.....	306
Figura 5.7	Generación de la Secuencia de Reconocimiento.....	308
Figura 5.8	Generación de condición de Parada.....	310
Figura 5.9	Formato de la transmisión.....	311
Figura 5.10	Diagrama de Tiempos del Maestro.....	317
Figura 5.11	Diagrama de flujos Firmware Maestro.....	319
Figura 5.12	Diagrama de Tiempos del módulo Esclavo.....	324
Figura 5.13	Diagrama de Flujos Firmware del Esclavo.....	326
Figura 5.14	Señal en forma binaria '10100100' en modo Asíncrono...	327
Figura 5.15	Formato del envío de datos recolectados por el módulo Maestro a la PC.....	329
Figura 5.16	Diagrama de tiempos de la transmisión de un byte con el Módulo USART.....	333
Figura 5.17	Diagrama de tiempos de la transmisión de mas de un byte con el módulo USART.....	333
Figura 5.18	Diagrama de Flujo del módulo de comunicación USART.....	335
Figura 5.19	Diagrama de conexiones del RS-232 con el Pic16F877A.....	338
Figura 5.20	Direccionamiento indirecto utilizado por el módulo de	

	control de memoria.....	341
Figura 5.21	Diagrama de Flujos del funcionamiento del Módulo.....	344
Figura 5.22	Diagrama de bloques del módulo Esclavo.....	346
Figura 5.23	Figura del circuito de monitoreo de un circuito de baterías.....	347
Figura 5.24	A) Forma de onda de la corriente que circula por las Baterías. B) formas de ondas del Voltaje de entrada AC y el Voltaje que recibe la batería.....	349
Figura 5.25	Circuito de fuerza de un Cargador de Baterías con una Inductancia, la batería recibe una corriente más DC pura.....	350
Figura 5.26	Circuito sin inductancia, la batería recibe pulsos de Corriente. ....	351
Figura 5.27	Circuito de tratamiento y amplificación de la señal De la Corriente.....	352
Figura 5.28	Circuito de Fuerza de un cargador, donde VBat es la señal que va a la tarjeta Esclava para su lectura.....	354
Figura 5.29	Circuitería de tratamiento de la señal de Voltaje de la línea de Baterías.....	356

Figura 5.30	Conexiones del LM334Z.....	359
Figura 5.31	Ajuste de la tasa de incremento de ISET, la Pendiente de cada gráfico es inversamente a RSET.....	360
Figura 5.32	Salida del circuito de la tarjeta transductora (jumper 2-3), que nos da una señal de 0 – 20mA.....	362
Figura 5.33	Circuito de entrada de la tarjeta transductora, que nos da una señal de 0 – 20mA.....	365
Figura 5.34	Función de Transferencia entre V02 y V01, TR1 es el parámetro que ajusta la pendiente De la curva.....	366
Figura 5.35	Tercera etapa de amplificación de la tarjeta Transductora 0-20mA.....	367
Figura 5.36	Circuito que fija el voltaje de referencia en el Pic Para la Conversión.....	372
Figura 5.37	Ciclos de conversión TAD.....	373
Figura 5.38	Diagrama de flujos de la operación del módulo A/D.....	375
Figura 5.39	Función de transferencia.....	382
Figura 5.40	Señal de corriente que muestrea el módulo esclavo a una Frecuencia de 19,5 KHz.....	384
Figura 5.41	Diagrama de flujos del módulo de muestreo y promedio e	

	señales analógicas.....	387
Figura 5.42	Esquema de conexiones del Sistema de Monitoreo, tarjeta .....	389

## INDICE DE TABLAS

Tabla I	Cuadro Comparativo cuatro tipos de Pic's gama 16F87x	135
Tabla II	Aplicaciones con los Pic16F877A.	137
Tabla III	Características del Pic16F877A	139
Tabla IV	Valores que toman los bits RP1:RP0 para acceder a los 4 bancos de memoria	150
Tabla V	Instrucciones del Pic16F877A	157
Tabla VI	Función de los bits del registro STATUS	163
Tabla VII	Función de los bits del registro OPTION_REG	164
Tabla VIII	Función de los bits del registro INTCON.	168
Tabla IX	Función de los bits del registro PIE1	169
Tabla X	Función de los bits del registro PIE2.	170
Tabla XI	Función de los bits del registro EECON1.	176
Tabla XII	Se muestra las funciones de cada pin de la Puerta A.	184
Tabla XIII	Se muestra las funciones de cada pin de la Puerta B	189
Tabla XIV	Se muestra las funciones de cada pin de la Puerta C.	192
Tabla XV	Se muestra las funciones de cada pin de la Puerta D.	196
Tabla XVI	Se muestra las funciones de cada pin de la Puerta E.	198
Tabla XVII	Función de los bits del Registro ADCON0	202

Tabla XVIII	Funciones de los bits del registro ADCON1.	203
Tabla XIX	TAD vs. máxima frecuencia de operación del dispositivo	208
Tabla XX	Bits de control del Registro T1CON	215
Tabla XXI	Bits de Control del Registro T2CON	217
Tabla XXII	Acciones del dispositivo cuando se recibe un byte según el estado de BF y SSPOV.	229
Tabla XXIII	Bits del control del registro TXSTA	241
Tabla XXIV	Bits del control del registro RCSTA	243
Tabla XXV	Baudios generados por el modo asíncrono con BRGH=1	247
Tabla XXVI	Indica las diferentes causas de Reset.	259
Tabla XXVII	Acciones de una transferencia de datos de acuerdo a los bits BF y SSPOV.	323

## INTRODUCCION

El presente trabajo trata de un sistema de monitoreo de líneas de carga de baterías. Diseño e implementación de la interfase serial RS-232 entre los Microcontroladores Pic y la PC enfocado a la adquisición de datos que intervienen en un proceso de formación de baterías mediante un monitoreo electrónico, con una comunicación serial entre los módulos electrónicos y la PC.

En la formación de baterías intervienen algunos parámetros que son determinantes para su vida útil, por lo que el monitoreo de baterías, tanto en el momento de su formación y durante los ciclos de carga y descarga que se producen durante su uso es un medio útil para llevar un control adecuado de estas. Teniendo el costo de una batería los sistemas de monitoreo se han vuelto muy útiles.



Este trabajo, se diseñó e implementó en la fábrica de Baterías Lux fabricantes de la marca alemana BOSCH debido a su alta producción, esta demandaba un sistema central que pudiera tomar lecturas de los principales parámetros en el proceso vía módulos electrónicos y luego transmitirlos a un software que

podiera analizar los datos y llevar un estadístico de baterías formadas o determinar un error en el proceso de carga.

# CAPITULO 1

## 1.- INTRODUCCION A LOS SISTEMAS DE MONITOREO

### 1.1 UTILIDAD DE LOS SISTEMAS DE MONITOREO.

Las baterías de plomo ácido que se usan en la actualidad para las aplicaciones de fuentes de alimentación interrumpibles (UPS) y de telecomunicaciones están muy lejos de las mejoras que hizo Planté de las celdas secundarias originales de plomo y ácido sulfúrico de Sinstede. A pesar de todas las mejoras logradas con los materiales que se usan en los electrolitos, separadores, placas y gabinetes, la introducción de las celdas VRLA (*Valve Regulated Lead Acid* – de plomo ácido con válvula reguladora) a comienzos de los 80 sólo confirmó que todavía seguía con vida la tecnología desarrollada fundamentalmente en la década de 1850.



Además, los métodos tradicionales empleados en el mantenimiento de las celdas de plomo ácido, la medición de la profundidad y el peso específico del electrolito, dejaron de ser prácticos con la introducción de la celda VRLA. Ahora el gabinete de la celda (ya sea en forma de 'vaso' o 'monobloc') venía sellado, con excepción de una válvula de liberación de presión. El electrolito pasó a ser un gel y el mantenimiento se redujo a la medición trimestral de la tensión y, con suerte, a una prueba de carga.

Sin embargo, de esto surgió una industria, en especial en los Estados Unidos, cuyo objeto era proveer medios 'conectados' de monitoreo electrónico de las baterías. Al comienzo consistía en la medición periódica de la tensión de cada una de las celdas en la cadena de baterías, a través de un largo conducto de cableado. Hay muchos sistemas en uso hoy día en todo el mundo que continúan haciéndolo con éxito.

No obstante, la celda podía sufrir degradación que quedaba oculta a la prueba de tensión solamente. La mejor prueba seguía siendo la prueba de descarga periódica. Pero, sucede que en los intervalos entre las pruebas trimestrales,

una celda puede tener una falla. Una celda VRLA en una UPS puede fallar en cuestión de pocos días tras la prueba de descarga de rutina. Las averías son causadas por los cambios químicos que tienen lugar en la celda ocasionados por los mismos ciclos de carga y descarga de la UPS. Son los ciclos que se producen incluso aunque el UPS nunca entre en línea. Asimismo, puede haber corrosión en todos los elementos de conexión de la cadena de baterías, internos a la celda o externos.

En consecuencia debía encontrarse un método que superara la prueba de tensión solamente. La resistencia o impedancia de origen de la celda se aisló como un indicador apropiado de la condición de la celda, indicando tanto la existencia de degradación química como de daño físico.

En la actualidad son varias las compañías en el mundo que suministran instrumentos portátiles o de conexión semifija o fija que proveen datos electrónicos para predecir con confiabilidad las fallas en las celdas.

## 1.2 OBJETIVOS.

- Diseño de un Software para visualizar el estado de la líneas de carga de baterías solicitadas por el operador como corriente, temperatura, horas de carga, amperios-horas, animación de líneas de carga.
- Diseño e Implementación de tarjeta de comunicación de datos entre PC y Pic mediante una interfase serial RS-232.
- Diseño e Implementación de tarjetas de adquisición de datos para los parámetros de temperatura, voltaje y corriente de línea.

## 1.3 ESTUDIO Y DESARROLLO.



Para la realización de este proyecto se necesitó un estudio que se inició en la fábrica de Baterías Lux, fabricantes de la marca Bosch en Ecuador. El estudio se enfocó en el comportamiento de las baterías durante los procesos de formación y los parámetros más importantes que influyen en estas. Los parámetros que se observan durante el proceso de formación son: corriente,

temperatura, voltaje, capacidad de reserva o amperios horas, horas de carga, densidad pero existen otros factores que intervienen en la formación pero dependen de la constitución interna de la batería al momento de su fabricación como la pasta de óxido de plomo que lleva una rejilla, la aleación de la que está hecha la rejilla, el número de placas positivas y negativas de cada celda, etc.

Son muchos factores al momento de la formación pero con los parámetros más importantes ya mencionados, se puede llevar un control adecuado del estado de la batería. El proceso de formación comienza estableciendo que línea de carga se va a utilizar y cuantas baterías se van a incluir en ella esto depende del voltaje de salida nominal del cargador de baterías de la línea, dividiendo el voltaje de salida para el voltaje de la batería que usualmente es de 12 Voltios obtenemos el número de baterías en la línea, por lo general son de 12 a 18 baterías de acuerdo al modelo. Las baterías se llenan de ácido a una densidad que la establece la fábrica, pero hay algunos procesos que usan ácidos a diferentes densidades a través del proceso, esto se explica en el capítulo 2, luego se las pone en una mesa de cemento o banco. Las baterías son conectadas en serie, terminal negativo de una unido con el terminal positivo de otra con conectores. Seguido de conectar en serie las baterías, se conectan las

terminales de salida del cargador a los extremos de la línea de baterías. Una vez conectadas las baterías se establece la corriente de carga, según el régimen de carga de la fábrica se establece la corriente. Existen cargadores computarizados que varían la corriente DC considerando no sobrepasar la temperatura máxima pero hay otros que tienen un control de corriente manual, con una perilla, que la mantiene la constante a través de toda la carga. Dependiendo del número de placas por celda se establece con el régimen de carga los amperios por placa que se necesitan para producir la reacción química necesaria para almacenar energía en ellas.

Luego de establecer la corriente, se debe establecer la capacidad de reserva de la batería, este parámetro depende del tipo de batería fabricada (constitución interna), se mide con amperios horas, que no es más que la cantidad de corriente entregada en una cantidad determinada de tiempo. Con cargadores de corriente variable el parámetro que toma más importancia son los amperios horas totales ya que la corriente siempre está variando y las horas de carga se ajustan hasta que se cumplan los amperios horas establecidos. Con cargadores de corriente constante las horas de carga son fijas hasta completar los amperios horas totales por lo que la capacidad de reserva la determinan las horas de

carga. La densidad de la solución ácida es uno de los parámetros medidos durante el proceso por los operadores, debe pasar de una densidad inicial durante la carga a una densidad final que lo establece la fábrica. El otro parámetro es la temperatura ya que esta por lo general no debe pasar de 60 °C ya que ocasionaría daños internos en la batería. Esta temperatura se puede dar por una excesiva corriente, cuartos de carga con poca ventilación, etc. El voltaje da la línea de baterías crece a lo largo de la carga con cierta relación a la densidad de solución ácida y es un indicativo que la batería está acumulando energía. Las lecturas de voltaje, corriente, horas de carga y amperios-horas físicamente se encuentran en los cargadores mientras que la temperatura y densidades es directamente en el cuarto de carga (En las baterías), cabe indicar que el cuarto de cargadores no se encuentran cerca por las condiciones corrosivas del ácido del cuarto de carga, por lo tanto el análisis y/o control del proceso tiene mayor dificultad por las distancias entre los puntos de las lecturas. Todos estos parámetros medibles intervienen en un proceso de carga y dichas lecturas se hacen por separado, con diferentes instrumentos y operadores, por lo tanto para tener el buen control y análisis de estos parámetros y del proceso surge la necesidad de tenerlos centralizados en una PC, en un sistema donde un software realiza consultas de procesos de carga, animaciones, estadísticos de producción, etc.

El desarrollo de este proyecto diseña e implementa un sistema que monitorea la formación de líneas de baterías. El sistema instalado consta de hardware que está formado por tres módulos, dos módulos que se encuentran en un cargador de baterías y que recoge los datos de carga periódicamente, los cuales son voltaje, corriente y temperatura (parámetros importantes en la formación de una batería), llamados módulos Esclavo y el otro módulo, llamado Maestro, es el que realiza la comunicación con los módulos esclavos y luego que obtiene los datos de todos, los organiza en un formato definido en el sistema y los envía a la PC para su posterior análisis. La otra parte del sistema es un software, que con los datos recolectados por el hardware se puede tener una visualización de las líneas de baterías en 3D con varios colores de acuerdo a los diferentes estados de la carga y muestra los parámetros de voltaje, corriente, temperatura, amperios-horas y horas de carga de cada línea, realiza curvas de carga, consulta en tablas los procesos de carga, cuadro estadístico de número de baterías formadas por mes, tabla de alarmas, archivos históricos de carga.



CIB-ESPOL

#### **1.4 CONTRIBUCION DEL SISTEMA DE MONITOREO A LA FABRICACION DE BATERIAS.**

En el mercado se encuentran diferentes diseños de cargadores que incluyen en su funcionamiento el control y monitoreo a través de una "Host PC", así podemos mencionar compañías como Bitrode Company y Firing Circuits cuyos diseños cuentan con un software de control y monitoreo que contribuyen a una formación eléctrica de batería de calidad que a largo plazo influye en el tiempo de su vida útil.

Los nuevos diseños de cargadores poseen circuitos de control completos y sistemas de monitoreo que proporcionan reportes de datos basados en el equipo de carga. La arquitectura de estos sistemas provee a los usuarios control de cualquier circuito desde cualquier lugar de una red local de computadoras. Los operadores pueden establecer en software un régimen de formación de una línea de baterías a través de programas que son definidos para cada tipo y modelo de batería.

Los programas que intervienen en cada formación tienen condiciones limitantes. Estas condiciones limitantes son monitoreadas a través del hardware del cargador y organizadas por el software de tal manera que el programa de formación se ejecuta en base a estos datos. Las variables o estas condiciones pueden escogerse a través del software tales como tiempo, voltaje, corriente, temperatura, amperios-horas, watts-horas y otros parámetros que de acuerdo al proceso de formación pueden ser incluidos en el monitoreo. Por ejemplo, el monitoreo de la temperatura ayuda a la compensación del calentamiento de las baterías durante el proceso, por medio del ajuste automático de la corriente.

Los circuitos que adquieren los datos pueden ser fácilmente monitoreados con el software y su actividad puede observarse en ambos formatos tabular o gráficamente. La adquisición de datos puede ser ejecutada según la especificación del usuario y su visualización puede realizarse en forma independiente o en conjunto permitiendo comparaciones entre variables.

Toda la información recolectada del proceso de formación se guarda en bases de datos como por ejemplo la de Acces, con lo que se facilita la realización de

reportes del proceso y luego permite verificar y tomar acciones sobre el proceso de formación y sus programas, en determinada línea de producción de baterías. De esta manera se explica la contribución de los sistemas de control y monitoreo en la fabricación de una batería, intrínsecos cada día en las características de los diseños de los cargadores de baterías.

# CAPITULO 2

## 2. INTRODUCCION A LAS BATERIAS.

### 2.1 REVISION DE LA FABRICACION DE BATERIAS.

#### *MOLDEO DEL PLOMO*

En un principio, todas las piezas del plomo se hacían a mano y en muchos casos se usaban moldes de una sola cavidad. Después de la Primera Guerra Mundial y ya entrada la década de los 20, las máquinas de moldeo y los moldes de varias cavidades substituyeron a los métodos y al equipo de producción más lentos. Los recubrimientos de corcho eran desconocidos. El humo de acetileno y los polvos de talco eran dos de los materiales que se usaban sobre las superficies de los moldes. El uso de humo de Acetileno se había desarrollado en un procedimiento detallado en el cual un recubrimiento de humo "duro" era

seguido por un recubrimiento de humo "suave". Algunos de estos recubrimientos tenían una duración de aproximadamente medio día de trabajo. El uso de polvos de talco era un proceso completamente independiente y diferente del método de humo de Acetileno. En un molde limpio y caliente, usando una gran bolsa de tela porosa, se hacía una aplicación abundante de polvos de talco a las superficies del molde entre cada vaciado para el moldeo de plomo. A fines de la década de los 20, la mayor parte de la industria ya estaba cambiando a una mezcla de carbón negro y arcilla de bentonita en agua o a los recubrimientos de rociado de molde con polvo de corcho usando silicato de sodio y agua como sus ingredientes líquidos esenciales. El recubrimiento a base de corcho se usa todavía, actualmente con muchas adiciones y con modificaciones de fórmula.

Los moldes para rejillas se hacían tal como se hacen actualmente, es decir, con un diseño de rejilla doble en cada cavidad del molde. Algunos se hacían con la cavidad montada en forma vertical, con la parte más larga en posición tal que el plomo se vaciaba por toda parte larga adyacente a las orejas de la rejilla. Otros moldes se montaban verticalmente a lo largo del lado más extenso del molde y el plomo se vaciaba por la parte superior de la cavidad. El desescoriado de la olla de vaciado del molde se hacía a mano, algunas veces con ayuda mecánica.

El plomo antimonial de la rejilla variaba entre 7% y el 12%, la mayor parte dentro del 9 % al 11 %.

Durante la primera década de los acumuladores automotrices, aparecían y desaparecían muchos diseños de rejillas después de una corta experiencia práctica de servicio. Además del diseño de alambre para rejilla horizontal y vertical, que hace que se formen pastillas de material activo en forma regular (Que todavía prevalecen en los diseños de rejillas actuales), existían en el mercado diferentes formas geométricas de varios tamaños. Tales diseños presentaban superficies de pastilla en forma cuadrada, romboide, triangular o hexagonal. Para justificar tales formas, se argumentaban una serie de cosas, incluyendo la fuerza mejorada de la estructura de la rejilla. Algunos argumentos llegaron todavía más lejos, afirmando que eran a prueba de deformación de las placas positivas. La experiencia práctica, sin embargo, por lo general no confirmaba los argumentos. No había control sobre el peso de la pasta para asegurarse de que el material no se desprendiera de manera prematura en servicio. Los sistemas de carga tenían pocos controles de voltaje para reducir o evitar sobrecargas. Por consiguiente, una causa común de falla de las baterías era la deformación de las placas positivas, que atravesaban las esquinas de los

separadores adyacentes y los agujeros oxidados de los centros del separador, sin importar el diseño de la rejilla.

Muchos de los que apenas se inician en la industria se han preguntado por qué tales piezas de plomo no se hacen a "troquel", un proceso bien desarrollado por medio del cual la producción es bastante elevada. La razón es que el troquelado tiende a ser poroso, teniendo algunas sopladuras, dan como resultado rejillas defectuosas que producen baterías de corta vida.

#### *Preparación de pasta y empastado*

En aquel entonces, los óxidos de plomo utilizados como materiales principales para placas positivas eran el plomo rojo ( $Pb_3O_4$ ) y el litargirio. En algunas ocasiones se añadían cantidades más pequeñas de litargirio ahumado o sulfato de plomo básico como "agente de volumen" para reducir el peso de la pasta. Las placas negativas por lo general utilizaban litargirio como material de óxido de plomo. Algunos fabricantes usaban aquí también un poco de litargirio ahumado y sulfato de plomo básico. El expansor, si acaso se usaba, consistía

inicialmente en sulfato de bario. Posteriormente, el expansor se preparaba "digestando" algodón o material de algodón de desecho de ácido sulfúrico al 50-60%, a lo cual en ocasiones se añadían sulfato de bario. Algunas plantas utilizaban polvo de madera o aserrín en lugar de los artículos de algodón mencionados en la fórmula de expansión anterior.

Las soluciones de la preparación iban de ácido sulfúrico a 1.100-1.300 de densidad relativa, mezclas de ácido y soluciones acuosas de sulfato de amonio en el rango de 1.050-1.200 de densidad relativa. El empastado se hacía a mano, una placa doble por vez sobre lona pesada, o varias placas a la vez sobre un tablero con papel por debajo para absorber el agua. Posteriormente, algunas de las compañías de baterías más grandes usaban máquinas de empastado en crudo, algunas de las cuales eran principalmente transportadores mecanizados de banda de lona con algo de calor aplicado. Una de las primeras máquinas de empastado incorporaba un tratamiento de inmersión en ácido para las placas. Sin embargo, pronto se comprobó que éste no era satisfactorio ya que la corrosión ácida estaba fuera de control y además se dificultaba el manejo posterior de las placas.

### *FORMACION, ENSAMBLE Y CARGA FINAL.*

Después del empastado, se daba a las placas alguno de los siguientes tipos de tratamiento:

- Se les colocaba en anaqueles en tanques de formación con ácidos sulfúricos diluido durante algunas horas y posteriormente se cargaban durante varios días, produciendo placas formadas negativas y positivas. En caso de que la superficie de las placas tuviera probabilidades de secarse con el aire mientras se pasaban a los tanques de formación, a dichas placas se les aplicaba una inmersión en agua.
- A las placas empastadas se les aplicaba de inmediato una inmersión en ácido y se guardaban una noche.
- Las placas empastadas se sumergían durante 16 horas en grandes tanques que contenían ácido sulfúrico. A tales placas algunas veces se les llamaban "placas decapadas".



CIB-ESPOL

Se intento llevar a cabo una producción limitada usando vapor sobrecalentado a bajo presión, el cual quitaba la humedad excesiva de la pasta con rapidez y "fijaba" el material activo. Esto, por lo general producía bastantes defectos de placa e incrementaba el plomo en las corrientes de aire en el departamento.

Otro método en las líneas de formación consistía en soldar las placas tratadas en elementos sin formar y después formar éstos en los tanques ácidos durante varios días.

El ensamble de la batería consistía en placas formadas en seco, y, como tales, las baterías ya ensambladas podían cargarse por varios días antes de su embarque o almacenaje. Dichas baterías podían embarcarse sin estar cargadas, para ser cargadas posteriormente por el distribuidor a su conveniencia. El otro tipo de ensamble consistía en elementos formados húmedos, los cuales, después del ensamble de la batería, se cargaban durante la noche y después se embarcaban a se almacenaban.

Los separadores, casi en su totalidad de madera de Cedro, eran tratados con soda cáustica diluida hirviendo y después enjuagados con agua hasta dejarlos libres de soda cáustica. Este tratamiento por lo general llevaba 24 horas.

### *ASPECTOS MISCELANEOS*

Aparecieron en el mercado algunas baterías de construcción especial apoyados en varios argumentos. Algunas baterías utilizaban separadores de celuloide perforados corrugado sin madera y placas negativas "sin expansores". Algunas veces, tales combinaciones se usaban también con electrolito de "gel", que se elaboraba añadiendo silicato de sodio al ácido de la batería. Por lo general, todas estas baterías "especiales" fallaban a través de los separadores perforados. Durante este período, se hicieron algunos intentos por aplicar materiales de plomo a las rejillas usando diferentes al de la mezcla usual de pasta de óxido de plomo. Uno de estos fallidos intentos consistió en hacer una pasta acuosa o ácida de peróxido de plomo ( $PbO_2$ ) que se aplicaba a la rejilla. Lo mismo se intento usando plomo pulverizado para fabricar las placas negativas, lo cual tampoco tuvo éxito. El peróxido de plomo pulverizado, seco, se fijaba a las rejillas aplicando toneladas de presión.

Al plomo pulverizado también se le daban el mismo tratamiento. Todos esos esfuerzos, después de la carga en ácido diluido, daban como resultado fallas porque, ya fuera el peróxido de plomo o el plomo pulverizado, propio comenzaban a separarse y a desprenderse de la estructura de la rejilla que lo sostenía. No se encontró método mejor que el de usar la mezcla de pasta de óxido de plomo y aplicarla a la rejilla. La electro-formación producía la máxima adhesión y rigidez del material al costo mas bajo.

En los primeros años, un fabricante de baterías produjo baterías troquelando rejillas con plomo antimonial en hojas en vez de fundir las rejillas en la forma convencional. Las rejillas troqueladas no tenían el diseño de material activo con pastillas, interconectadas, sino aperturas o agujeros rectangulares sencillos dispuestos en un diseño que daba la apariencia final de una rejilla normal. La pasta de la batería común y corriente rellenaba los agujeros de la rejilla troquelada para las placas negativas y positivas respectivamente. Las baterías ya terminadas tenían muy poco rendimiento. El material activo formado en los agujeros de la rejilla troquelada se aflojaba y la batería fallaba. Esto creaba la

condición de que existían, si acaso, muy poca adhesión o contacto real entre el metal de la rejilla y el material activo de los muchos agujeros. Algunas áreas de material activo al parecer se contraían o se desprendían de las superficies de la rejilla de metal. Desde luego, prácticamente sin adhesión entre el material activo y las rejillas, no había rendimiento o suministro de corriente por parte de la batería.

A principio de la década de 1920 apareció en la industria del acumulador, el primer separador no hecho de madera, un producto completamente sintético. Aunque se pensó que una batería con tales separadores tendría un excelente rendimiento eléctrico, el resultado fue el opuesto. Los separadores de madera para acumular superaban en rendimiento a los acumuladores de separadores no hechos de madera. Esta situación puso en evidencia el hecho de que los separadores de madera no sólo servían a los fines normales de la batería, sino que de alguna manera también mejoraban el rendimiento eléctrico de la misma. De ahí, se llegó a la conclusión de que una de las propiedades adicionales de los separadores de madera para baterías era contribuir a mejorar el funcionamiento de los expansores en placa negativa comenzaron a incluir productos de madera tales como el aserrín o el polvo de madera. Tales

adiciones a la pasta negativa produjeron el rendimiento deseado para el uso de los separadores sintéticos. De esta manera, los productos de madera pasaron a formar parte de la fabricación de baterías.

Los primeros separadores sintéticos para baterías hicieron posible los primeros acumuladores de carga seca a principios de los años 20. Dichas baterías eran más caras debido a que (1) los separadores sintéticos eran más caros que aquellos de madera y (2) a la carga en seco de las placas eran una operación costosa.

Desde un principio, las baterías automotrices eran armadas, de tal manera que en cada celda había siempre una placa negativa más que placas positivas. De esta forma, las placas exteriores de un elemento son siempre placas negativas, p. Ej. , un elemento de siete placas tiene cuatro placas negativas y tres positivas. Durante el periodo inicial, había un tipo de baterías que se armaba exactamente a la inversa en lo que respecta a la estructura del elemento. Esto significa que un elemento de siete placas tenía cuatro positivas y tres negativas. A dicha configuración se le llamaba de "zonas laterales positivas". En dichas

baterías, ya en servicio efectivo, las dos placas positivas exteriores por lo general se deformaban y hacían corto a través de los separadores adyacentes. Las superficies exteriores del material activo de las placas exteriores por lo general permanecían en su lugar y era de una textura áspera y consistente. El material activo de la parte inferior de las mismas placas positivas mostraba las características de desprendimiento propias del servicio. Las demás placas positivas interiores mostraban también el mismo desprendimiento propio del servicio. Después de algunos años en el mercado, se discontinuó la configuración de "zonas laterales positivas", para resurgir en parte con la celda actual, que tiene el mismo número de placas, y en donde una placa exterior es negativa y la otra placa exterior es positiva.

Tiene interés histórico la descripción de la primera batería automotriz de producción masiva, para arranque eléctrico del motor. Perteneció a un auto fabricado en los Estados Unidos anunciado a finales de 1911 para el modelo 1912. La batería de este carro fue descrita posteriormente por el fabricante de la batería, en folletos, de la siguiente manera:

“La batería 1912 tenía 12 pequeñas celdas, cada una compuesta por tres placas en un pequeño recipiente de caucho (hule) duro con una tapa de caucho (hule) suave. No se utilizaba ningún compuesto sellador. Las celdas ensambladas en cuatro grupos de tres placas cada una, estaban conectadas en serie para el arranque a 24 volts y en paralelo para el alumbrado y la ignición a 6 volts. Esta batería, que pesaba alrededor de 67 libras, se había usado en años anteriores y en conexión con otros dispositivos que requerían de altos volúmenes de descarga”.

Hasta 1920, los sistemas eléctricos automotrices eran de 6 o 12 volts, o una combinación de encendido de 12 volts y alumbrado y carga de 6 volts. Esta última combinación incluía carga en serie paralela. Si la conexión de una de las secciones de la batería de 6 volts fallaba y no hacía el contacto adecuado, esa parte de la batería no se cargaba. Tres celdas se cargaban por completo y tres se descargaban. Para 1925, todos los autos contaban con sistemas de 6 voltios, a excepción de los de un fabricante del sistema eléctrico de 12 voltios.

En 1931, se introdujo en los Estados Unidos el primer auto sin dispositivo de arranque de motor de manivela para ocasiones de emergencia. Esta nueva característica pronto se aplicó a todos los autos nuevos. La ausencia de la manivela fue señal de la aceptación de la batería como parte confiable del automóvil.

Las conexiones directas "a través de la pared" entre las celdas de la batería se usaron por primera vez en 1935. Esta configuración se usaba sólo con cajas de caucho (hule) duro y era el equipo de batería estándar usado solamente por un fabricante de autos y por un fabricante de camiones. También se utilizaba en una batería de alto rendimiento para aviones. Una ciudad grande usaba dicha configuración en baterías de uso pesado para sus autobuses urbanos. Todos aquellos que en aquel entonces se dedicaban a dar servicio a baterías, se oponían firmemente a los "conectores enterrados". Una de las principales objeciones a dicha configuración era la dificultad para obtener la lectura de las celdas individualmente. Aun cuando no se detectaron fallas de servicio, la configuración se discontinuó después de un año o dos, en vez de tratar de superar el continuo rechazo del personal que daba servicio a baterías. Sin duda alguna, esa configuración era muy avanzada para su época.

Durante los primeros años de la Segunda Guerra Mundial, el Comité de Producción de Guerra de los Estados Unidos se encontró con que había escasez de antimonio para rejilla de baterías. Las baterías eran un artículo indispensable para fines militares. Por ello el Comité dispuso que todas las rejillas para baterías fueran de un 7% de antimonio-Plomo. En aquel entonces, la cantidad mínima de antimonio utilizada por cualquier fabricante de batería estaba alrededor del 7%, aunque algunos usaban porcentajes hasta del 11 %. A algunos fabricantes del Comité les dio un permiso especial para usar más del 7 % en las rejillas, ya que estas compañías no tenían ni el "know-how" ni el equipo necesario para moldear rejillas con un contenido menor de antimonio.

El moldeo de rejillas usando aleaciones de plomo-calcio-estaño (Pb-Ca-Sn) para baterías libres de mantenimiento es un indicativo del progreso que la industria ha tenido en los últimos 30 años. Hoy en día muy pocos fabricantes de baterías usan antimonio en proporciones tan altas como el 7 %, la mayoría usa el 3 % o menos. Las rejillas de Pb-Ca-Sn están siendo moldeadas con máquinas, sin grandes problemas. Aún cuando a todo esto, las rejillas actuales para baterías

se producen a una velocidad de moldeo de hasta 20 golpes por minuto, un índice mucho mayor que cualquier índice de molde de la Segunda Guerra Mundial, cuando sólo algunos fabricantes podían alcanzar los 14 golpes por minuto.

La completa automatización de la producción de baterías automotrices ha sido difícil de alcanzar y ha llegado con retraso a la industria de la manufactura de batería, pero los problemas se están solucionando rápidamente y se encuentra en puerta un proceso sustancialmente continuo que va del plomo en lingotes hasta el acumulador terminado de manera directa y con pocas interrupciones esenciales.

## **2.2 FORMACION Y CARGA DE BATERIAS.**



Para esta operación no se requieren materias primas. Las baterías deben venir de la línea de ensamble listas para colocarse en la línea de formación o carga.

### 2.2.1 EQUIPO REQUERIDO.

1.- Suministro de energía de corriente directa. Este debe ser, de preferencia, de algún tipo de diseño de rectificador en seco, probablemente de silicio. Los voltajes de salida pueden variar de 60 a 250 voltios máximo, lo cual permitirá colocar aproximadamente de 6 a 28 baterías de 6 voltios en cada línea de carga o la mitad de ese número en rectificadores de capacidad de 20 a 25 amperes para cada circuito de carga. Cada unidad debe incluir un interruptor principal y un amperímetro con una escala de 25 amperes de capacidad aproximadamente, graduada en divisiones de  $\frac{1}{2}$  amperes. La unidad deberá también contar con mecanismos manuales para controlar el régimen de carga  $\frac{1}{2}$  Amper del régimen deseado.

Existen rectificadores que cuentan con un bulbo especial para cambiar el suministro de entrada de energía de corriente alterna a corriente continua. Este tipo es menos efectivo que el tipo seco y su uso ha disminuido considerablemente en los últimos años.

También se han usado unidades moto generadoras, pero están son ahora menos populares que en el pasado. Los cargadores de corriente constante se han vuelto populares y son los más efectivos.

Sin importar el tipo de unidades de carga eléctrica que se utilicen, éstas deberán ubicarse en la planta de tal forma que sea prácticamente imposible que los vapores del ácido de las baterías en carga lleguen a las unidades. Si los vapores llegan a las unidades de carga, es de esperarse una severa corrosión de las partes metálicas y esto puede ocasionar problemas. Algunas plantas tienen los cargadores en un cuarto separado que cuenta con presión de aire positivo generada por un suministro de aire distante. En climas tropicales, las baterías pueden ubicarse fuera del mismo edificio de la planta.

Con tal ubicación, solo es necesario contar con la protección mínima de un techado para evitar la lluvia. En cuyo caso, las unidades de carga ubicadas dentro del edificio se conectarán con cables que pasen a través de un pequeño agujero en la pared.

2.- Para la producción programada de baterías, se requieren suficiente conectores de cables con sujetadores para efectuar las conexiones eléctricas entre batería y batería. Se pueden usar también flejes de bronce de aproximadamente 1 pulgada (2.5 cm.) de ancho y 12 – 15 pulgadas (30-35 cm.) de largo. Cada extremo de los flejes tiene una hendidura en “V” de 90 grados. Para usar el fleje, la hendidura en “V” se coloca al lado del borne terminal de una batería (el borne de manera perpendicular en la hendidura). El otro extremo del fleje se conecta de la misma forma al borne terminal de polaridad opuesta de una batería adyacente. Las baterías adyacentes deberían estar espaciadas a una distancia menor el largo del fleje, de modo que exista siempre tensión en el fleje.

3.- Se requieren varios hidrómetros de vidrio tipo jeringa para baterías con diferentes escalas de flotador para densidad relativa, algunos con una escala de 1.000 a 1.150 de densidad relativa. Se sugiere que dichos flotadores sean de un tamaño mayor al de aquellos que por lo general se encuentran en las estaciones de servicio para baterías. Se recomienda una longitud de flotador de 5 – 6 pulgadas (13 – 15 cm.). La facilidad de lectura es muy importante, de la

misma forma, es posible obtener una mayor precisión con un hidrómetro largo, un factor que vale la pena considerar al producir un producto de calidad. Es también importante mantener limpias todas las partes de vidrio del hidrómetro. El lavado con nafta y jabón son procedimientos que no deben pasarse por alto.

4.- Para revisar tanto la temperatura de mezclado del ácido como la de la batería, se utilizan algunos termómetros de mercurio en vidrio (o su equivalente) de aproximadamente 6 – 7 pulgadas (15 – 18 cm.) de largo, con una escala de 0 a 200 grados F con divisiones de por lo menos 2 grados.

5.- Se requiere varios voltímetros pequeños portátiles de corriente continua, con escalas dobles de 0 - 3 y 0 – 15 voltios. Si se utilizan instrumentos de una sola escala, el instrumento con escala baja deberá ser de 0 - 3 voltios y el instrumento con escala alta deberá ser de 0 – 20 voltios. Las divisiones más pequeñas de la escala baja deberán ser de por lo menos 0.5 voltios. Deberán ser de bastante utilidad los voltímetros o multímetros digitales calibrados. Son resistentes, fáciles de leer y tan precisos como el de tipo análogo pero más caro.

### **2.2.2 INSTRUCCIONES DE CARGA.**

Las baterías que lleguen a este departamento deberán estar bien llenadas con ácido de manera uniforme, y con poco o nada de ácido sobre las superficies exteriores de la batería. Debe realizarse una inspección para asegurarse de que no existe celda alguna sin ácido y celdas con un bajo nivel. La temperatura del ácido al momento del llenado no deberá exceder los 125 ° F (52 ° C).

Todas las baterías que van a ser cargadas por formación deberán ponerse a cargar tan pronto como sea posible después del llenado, excepto en condiciones de emergencia, tales como bajas corrientes o fallas de la misma, en cuyo caso las baterías deberán ponerse a cargar tan pronto se restablezca el suministro de energía.

Esto es especialmente importante con baterías hechas con separadores de papel impregnado de plástico, donde podría presentarse un agujeramiento de la separación debido a una larga espera entre el llenado y el inicio de la carga.

Las baterías deben colocarse en la mesa de carga de tal modo que exista espacio para ventilación entre los mismos. Esto permite que las baterías alcancen la temperatura mas baja durante la carga.

La cantidad de corriente eléctrica necesaria para formar placas en una batería se determina por medio de una cierta ley de la naturaleza inmutable, postulada inicialmente por un inglés llamado Michael Faraday y conocida como la ley de Faraday, encontró que la cantidad de energía eléctrica requerida para generar un cambio electroquímico en un metal, tal como el plomo, estaba relacionada con el peso equivalente de ese metal y que, en el caso del plomo, por ejemplo, éste requiere de 118 amperes- hora por libra para cambiar el plomo del oxido de plomo al plomo del peróxido de plomo (estado formado). En la práctica, se necesita más energía para resarcir las perdidas resultantes de la generación de gas y calor. Sin embargo, se pueden lograr muchas formaciones de placa

usando 125 amperes – hora por libra de material activo positivo por celda. La formación con el llenado de dos activos requiere entre 160 y 170 amperes – hora por libra, y los procesos de llenado de un ácido o de una carga requieren aproximadamente una tercera parte mas que le llenado de dos ácidos.

Los regímenes de amperes deben revisarse a intervalos regulares para asegurarse que los regímenes se mantengan adecuadamente. Cada circuito de baterías debe revisarse con respecto a la polaridad correcta, al momento de conectar el circuito de carga. Es bueno también verificar todas las baterías con un voltímetro dentro de la primera hora de carga para detectar cualquier batería defectuoso o que esté en corto.

El nivel del electrolito debe cubrir las partes superiores de los separadores durante el programa de carga. Al principio de cualquier programa nuevo de carga, que incluye nuevos tamaños de batería, nuevos tamaños de placa o cambios en cualquiera de los procesos de batería, debe usarse una celda o batería "piloto" para vigilar el avance de la formación, hasta que la experiencia

uniforme indique que ya no se requiere dicha atención. Esta práctica debe observarse tanto para una planta nueva como para una ya establecida.

Debe también recordarse que en ocasiones puede ser necesario vigilar una batería durante 6-10 horas antes de la terminación programada de la carga. En algunas ocasiones, especialmente con ensambles de placas más delgadas, las placas positivas pueden encontrarse casi totalmente limpias (con poco o ningún sulfato blanco en la superficie) en ese momento. Este es el criterio a seguir para la terminación de la formación. Los ensambles de placa más gruesa, tales como aquellos para camiones y autobús requieren el tiempo de carga completa y tal vez un tiempo adicional.

Algunos fabricantes de baterías prefieren discontinuar la carga de formación antes de que las placas positivas estén completamente libres de sulfato blanco en la superficie, con el fin de mejorar la vida de desprendimiento de las placas positivas en las baterías sujetas a descarga profunda. Hasta un 10 o 15 por ciento de sulfato en la superficie se deja de manera deliberada. En las placas



CIB-ESPOL

negativas siempre se limpian totalmente de todo sulfato que se encuentre en la superficie.

El tipo de programa de carga que debe seguirse en cualquier planta, es decisión de la gerencia. Se pueden poner en operación una amplia variedad de programas con resultados que implican costos de mano de obra, costos de equipo, superficies del piso del departamento, costos de la corriente de carga y algunas características de la batería final ya cargada. Algunos proveedores de óxido para batería pueden proporcionar las densidades de llenado de ácido recomendadas el igual que detalles de carga que deben usar los fabricantes de baterías al formar los mismos, particularmente si el proceso se lleva a cabo en las cajas de batería. . Después de algo de experiencia en la planta, el proceso deberá ser perfeccionado y refinado de modo que solo se requiera un mínimo de ajuste final en la densidad relativa del ácido. De la misma forma, excepto en circunstancias especiales, no se requerirá revisar la terminación de la carga (placas positivas limpiadas adecuadamente) al final del programa de carga.

Una herramienta muy útil para determinar la terminación de la carga es la "prueba de voltaje de cadmio". El cadmio es un metal similar al plomo en apariencia y puede obtenerse en forma de barra fundida de aproximadamente 25 de pulgada de diámetro y 2 pulgadas de largo. Siempre se toma una lectura del cadmio cuando la batería se encuentra casi al final de la formación. La lectura se toma insertando la barra de cadmio dentro del ácido que se encuentra por encima de los separadores y midiendo el voltaje entre el cadmio y el borne positivo, para obtener el voltaje de las placas positivas, y entre el cadmio y el borne negativo, para medir el voltaje de las placas negativas. Los voltajes obtenidos varían con el régimen de carga y con la temperatura, pero es posible obtener una lectura bastante precisa del estado de carga de las placas individuales.

Las buenas lecturas de cadmio positivas son de 2.40 voltios o más, las buenas lecturas de placa negativas son de más de  $-0.25$  voltios para un voltaje de celda de 2.65 voltios o más al final de la carga cuando las baterías se encuentran todavía en carga. Nótese la suma algebraica.

### 2.2.3 TIPOS DE CARGA.

#### *CILMA TEMPLADO.*

Hay dos diferentes programas o sistemas utilizados actualmente para formar baterías en las cajas, y cada uno será descrito a continuación:

#### *METODO DE LLENADO DE DOS ACIDOS O DE "DOS CARGAS".*

La carga de formación normal para un sistema de llenado de dos ácidos requiere de un tiempo de carga total de aproximadamente 36 horas, de las cuales 24 horas son a un régimen de amperaje alto, y las 12 restantes a un régimen de amperaje bajo. A continuación se proporcionan los detalles esenciales para este sistema.

- 1.- Primer llenado del electrolito, ácido de 1.065 – 1.1180 de densidad relativa.
- 2.- Carga de formación, 24 horas a régimen alto, 12 horas a régimen bajo.
- 3.- Electrolito al final de la formación, ácido de 1.140 – 1.220 densidad relativa.

4.- Vacíe o descargue el electrolito. Consérvelo, ya que se convierte en ácido reutilizable y se maneja como tal.

5.- Segundo llenado del electrolito, ácido de 1.400 – 1.290 de densidad relativa.

6.- Carga de refuerzo durante 6 horas a régimen bajo. La densidad relativa al final de este tiempo deberá ser de 1.255 – 1.285, según lo dicte la especificación corregida a 80 °F (27 °C) y el nivel a la altura correcta.

Si en el punto (1) se usa la densidad mínima, se verá que en el punto (3) la densidad resultante se encontrará en la escala más baja. De esta forma, debe usarse la escala más alta listada en el punto (5). O, si se aplica el extremo opuesto de las escalas, entonces en este caso se aplicaría el valor más bajo del punto (5).

Los valores exactos de ácidos que debe usar una planta dependen de cosas tales como las condiciones de la planta y los tamaños y las cantidades de los tipos de baterías programados. Esto es algo que sólo la gerencia local puede decidir.

Vale la pena recordar, como respecto a la carga de formación, que las placas positivas se forman mejor en ácido diluido de 1.075 – 1.225 de densidad relativa aproximadamente, mientras que las placas negativas se forman mejor en ácidos concentrados, alrededor de 1.200 – 1.300 de densidad relativa. Sin embargo, estas últimas no son tan sensibles o vulnerables a la concentración de ácido como las positivas.

Pueden existir ciertas condiciones locales que hagan aconsejable seguir procedimientos diferentes debido al costo o a otros motivos.

La experiencia ha demostrado que el programa de carga de régimen de amperes, es un programa práctico, considerando el costo de equipo, la superficie requerida de piso de la planta, el tiempo de material en proceso, mano de obra, etc. Sin embargo, es concebible que si hubiese una amplia superficie de piso disponible y el tiempo del material en procesamiento no fuera tan importante, se podrían reducir los regímenes de amperes, prolongando de esta forma el tiempo para carga de formación. Considerando una batería dada

que requiera de 6 amperes durante 24 horas y de 4 amperes durante 12 horas (6 amperes x 24 horas =144 amperes – hora y 4 amperes x 12 horas = 48 amperes – hora) lo cual daría 192 amperes – hora (144 + 48 = 192). Sobre esa base, es posible cargar las baterías durante 48 horas a 4 amperes (48 x 4 = 192 amperes – hora) y tener baterías completamente formadas. En este último caso, se requiere de más horas, específicamente 12 horas más. Esto significa que el equipo de carga y la superficie del piso se ocupan por más tiempo. La calidad del producto final en cada una sería la misma.

El caso es que la carga de formación puede hacerse con cualquier programa de régimen de amperes razonables siempre y cuando se usen los amperes – hora requeridos, y también que la temperatura de la batería durante el tiempo de carga no exceda aproximadamente los 130 ° F (54 ° C). Podría ser que un programa tal resultara provechoso, tal vez durante un fin de semana, cuando al mismo tipo de batería se le podría aplicar un régimen de carga de 3 amperes durante 64 horas (64 x 3= 192 amperes - hora) y tenerlo listo para las operaciones del lunes por la mañana.

Lo anterior ilustra un tipo de ajuste que es posible hacer a los programas si se entienden perfectamente los detalles de procesamiento en su totalidad. Sin embargo, esto es algo que los novatos deben evitar ya que la inexperiencia con procesos desconocidos puede ser costosa y desastrosa en lo que respecta a la calidad.

Una vez que las baterías han concluido la carga de formación de aproximadamente 36 horas, se vacía el ácido de las baterías y se les pone ácido más concentrado. Este es el segundo ácido del sistema de llenado de dos ácidos. La densidad relativa real usada en el segundo llenado de ácido tendrá que ser determinada por la planta, dependiendo de las condiciones locales, tales como los tamaños de batería, los procesos de placa, etc., que fueron utilizados.

Estos detalles han sido listados en un resumen de procesos anteriores. La descarga y el llenado de ácido de las baterías pueden hacerse manualmente o por medio de maquinaria de descarga y maquinas de llenado de ácido. El ácido

vaciado o descargado se convierte en ácido reutilizable, que se regresa al sistema de ácido principal, en donde se utiliza de nuevo.

Las baterías, después del segundo llenado de ácido, se ponen a cargar a régimen bajo, (al mismo régimen utilizado en la carga anterior de 12 horas) durante seis horas. Si todos los pasos se han seguido correctamente, el nivel y la densidad relativa del ácido deberán estar en proporciones aceptables y las baterías estarán listas para la colocación de los tapones de ventilación y el lavado. La carga de seis horas (denominada "carga de refuerzo") sirve exclusivamente como un método adecuado para igualar rápidamente las concentraciones de ácidos en las celdas de la batería. La carga de seis horas no añade nada a las características de la batería, porque las placas ya se encuentran completamente cargadas y tienen las características necesarias para la batería final.

### *METODO DE LLENADO DE UN ACIDO O DE "UNA CARGA".*

Este método para formación de placa de batería implica el llenado de un solo ácido en la batería ya ensamblada. La densidad relativa del ácido para llenado es mayor a la del ácido requerido para el llenado inicial del sistema de dos ácidos, comentado anteriormente. El objetivo final del sistema de un ácido es el terminar la carga de formación con una densidad relativa de 1.260 - 1.285 a 80 ° F (27 ° C) y con la altura de nivel de líquido deseada en todas las celdas. Los regímenes de carga son menores a aquellos especificados para el sistema de dos ácidos y continúan por lo menos durante 72 horas. Existen algunos fabricantes que creen que deben tener un tiempo de carga más corto por razones inherentes a las condiciones propias de su planta. Puede usarse un tiempo de carga más corto, pero no se recomienda debido al riesgo que implica. Se requerirían regímenes de descarga mayores, lo cual daría como resultado temperaturas más altas en la batería. En tales casos, una vez que las baterías han recibido los amperes-hora especificados, las superficies de la placa positiva tendrán cantidades variables de sulfato blanco. Dicho sulfato blanco es casi imposible de eliminar, aun con una cantidad considerable de corriente a tales baterías pueden o no dar resultados que indiquen superficies de placa positivas con sulfato blanco. Los resultados de la prueba eléctrica de cualquier



procedimiento de formación de "una carga" por lo general se encontraran de un 10 a un 15 % por debajo de los resultados obtenidos con el programa de formación de "dos cargas". Aun con tales resultados, que no son precisamente favorables, muchas baterías de "una carga" han dado un buen servicio automotriz. El tipo de condiciones internas de la batería y los resultados de prueba de batería aceptable a sus clientes son una decisión que corresponde al fabricante individual. Para esta presentación, se dan los datos para tiempo de formación de 72 y 96 horas. Sin embargo, pueden usarse periodos de carga más largos con regímenes de cargas todavía más bajos, siempre y cuando se use la energía eléctrica requerida en amperes – horas.

Una vez que la batería se llena con el ácido de densidad relativa seleccionada, aumentará la temperatura de la batería. Existe una reacción química entre el ácido sulfúrico y los materiales activos de la placa que genera un calor llamado "calor de reacción". Debe existir un período de espera de por lo menos una hora entre el llenado de ácido y las puestas en carga de las baterías. Esto da tiempo para que disminuya un poco la temperatura de la batería. Todas las baterías deben ponerse a cargar dentro de las cuatro horas siguientes al llenado

de ácido. Una temperatura de 150 ° F (36 ° C) es considerada el máximo para las baterías al inicio de la carga.

La densidad relativa real del ácido usado para el llenado de la batería tendrá que determinarse por ensayo. Se encontrará que cada tamaño de batería requerirá de una cierta densidad relativa inicial para terminar con las condiciones finales requeridas. Los tamaños de baterías que tengan elementos que ocupen las celdas de la caja por completo requerirán un peso de llenado ligeramente inferior al necesario para un tamaño de batería que tenga elementos que ocupen sólo de una manera parcial o espacio de las celdas del contenedor. Este último tamaño requerirá un ácido de densidad relativa mayor. Como sugerencia, se ofrecen cuatro diferentes densidades relativas para llenado de ácido: 1.230, 1.235, 1.240 y 1.245 a 80 ° F (27 ° C). Es evidente que estas cuatro densidades relativas deberán prepararse siempre con cuidado, ya que los valores tienen tan sólo una diferencia de 0.005 de densidad relativa entre sí.

Al principio, cada tamaño de batería debe probarse seleccionando una concentración de ácido y anotando cuidadosamente de densidad relativa final una vez que la batería está fuera de carga. Si la concentración final se encuentra consistentemente por encima del valor de densidad relativa requerida, la próxima prueba con el mismo tamaño de batería, deberá hacerse llenando el mismo con ácido de una concentración menor. Para el caso opuesto, si la prueba inicial da como resultado una densidad relativa que se encuentre consistentemente por debajo del límite, la siguiente prueba requerirá un ácido de llenado de una densidad ligeramente mayor.

Se encontrará que las baterías que han finalizado su carga de formación y se encuentran fuera de carga, casi siempre están bajos en los que respecta al nivel de líquido de la celda. Para aumentar el nivel a la altura requerida, se recomienda usar ácido sulfúrico de densidad relativa de 1.260 a 80 °F (27°C).

El plomo rojo reduce la cantidad de corriente carga (amperes – hora) requerida para completar la carga de formación. Para baterías destinados a servicio de energía automotriz, como el del caso de carros de golf, etc., en donde son

normales los ciclos profundos de descarga, no se usa plomo rojo en la pasta positiva. Esta carga mejora la vida cíclica en comparación con la pasta positiva para otras baterías. Las baterías sin plomo rojo en la pasta positiva requerirán más carga (amperes – hora).

### *CLIMA TROPICAL*

Para baterías de clima tropical, a falta de recomendaciones por parte del proveedor de óxido, las siguientes han demostrado ser satisfactorias:

- Llenado de electrolito, de densidad relativa de 1.175 – 1.185 aproximadamente.
- Carga de formación, 24 horas a régimen alto, 12 horas a régimen bajo.
- Electrolito final, de densidad relativa de 1.210 – 1.230 corregida a 80 °F (27°C).

Estas baterías no deberán requerir carga de esfuerzo. Las baterías de clima tropical tienen una densidad completamente cargada de 1.210 – 1.230 de densidad relativa, que tiene aproximadamente, 0.050 menos de densidad

relativa que aquélla de las baterías de clima templado. Esta condición se da porque en clima más caliente no se tienen los problemas para el encendido del auto que se tienen a temperatura fría y la densidad del ácido más baja prolonga la vida de la batería.

De nuevo, al igual que en el caso de las baterías para clima templado, la densidad correcta de llenado del ácido para diferentes tipos de baterías variará debido a las diferentes tipos de baterías variará debido a las diferentes relaciones entre la cantidad de material activo de la celda y el volumen total de la misma.

Como podrá notarse solo se da un programa de formación para baterías de clima tropical. Como estas baterías tienen una densidad relativa final más baja, tales baterías completan fácilmente la formación con el llenado de "una carga" y un programa de carga de dos regímenes en 36 horas. No se requiere vaciar el electrolito.

## CLIMA TROPICAL Y TEMPLADO

Ya casi al final de los programas de formación en baterías de clima templado y tropical, puede que sea necesario aumentar el nivel del electrolito con el líquido requerido, ya sea agua o ácido, dependiendo del peso de la batería, etc., de modo que la batería ya terminada se encuentre en el nivel correcto. La densidad final deberá encontrarse dentro de los siguientes límites:

Para clima templado, 1.255 – 1.285 de densidad relativa corregida a 80°F (27°C). Para clima tropical 1.210 – 1.230 de densidad relativa corregida a 80°F (27°C).

Por lo general, con cajas de plástico de colores claros que se someten a programas de formación, si es posible, el ácido del proceso debe mantenerse lejos de las superficies de plástico. El ácido que se encuentra sobre los acumuladores durante la formación puede dejar áreas decoloradas que le darán una apariencia exterior pobre.



CIB-ESPOL

En caso de que la densidad relativa no se encuentre en el rango correcto, esto se corrige sacando algo del electrolito de las celdas que requieren atención y sustituyéndolo con agua si la densidad relativa es muy alta; o agregando ácido de aproximadamente 1.380 – 1.400 de densidad relativa si el electrolito es muy bajo. La experiencia o práctica dictará las cantidades necesarias para cada operación. Deje cargar la batería a régimen bajo durante 2 horas aproximadamente y verifique la densidad relativa de nuevo. Si éste se encuentra en la escala adecuada, la batería está terminada y lista para la siguiente operación.

A las baterías que hayan permanecido por mas de 24 horas después de la terminación de la formación sin haberles verificado la densidad relativa , se les debe aplicar una carga de recuperación de 1 hora a régimen de formación bajo, por cada 24 horas que hayan permanecido en esta condición. Después de esta carga de recuperación, se puede tomar las lecturas de densidad relativa y se siguen los procedimientos normales.

En caso que se interrumpan los programas de carga de formación, el programa debe reiniciarse tan pronto como sea posible de modo que a cada circuito de acumuladores se le apliquen los amperes-horas requeridos.

Coloque los tapones de ventilación correctos en los agujeros de la tapa, asegurándose de que todos queden bien en su lugar. Lave las baterías para eliminar de la superficie cualquier ácido con un flujo moderado de agua de manguera. Retire el exceso de agua de la superficie de las baterías con una manguera de aire comprimido. Esta operación en su totalidad deberá hacerse de modo que no quede nada de agua o ácido sobre la superficie de las baterías. Esto es importante si se les va a aplicar pintura a las baterías inmediatamente. La pintura no se adherirá a una superficie húmeda. De la misma forma, recuerde que las gotas de ácido que se encuentren sobre la superficie de la batería no se secarán por evaporación.

*COMPARACION DE METODO DE LLENADO DE "UNA CARGA" vs. METODO DE LLENADO DE "DOS CARGA"*

*LLENADO DE DOS CARGAS*

Ventajas

La formación de la batería se produce más rápido, una vez cada 48 horas. La formación se lleva a cabo fácilmente, sin necesidad de usar plomo rojo en la pasta. Se necesita una mínima cantidad de espacio de piso. Se obtiene un óptimo rendimiento eléctrico de la batería. Alta temperatura durante la carga. Por lo general no existen problemas durante el verano.

Desventajas

Operación extra de llenado de ácido. Se necesitan operación y equipo de vaciado. Severa corrosión ocasionada por el ácido sulfúrico y vapores en el piso, edificio, etc. Se requiere de mucha ropa protectora de goma (hule) para los operarios.

## LLENADO DE UNA CARGA

### Ventajas

Sólo se requiere de un llenado de ácido. No se requieren equipo o procedimiento de vaciado. Mucho menos efectos corrosivos de ácido sulfúrico en el área circundante. Se necesita menos ropa protectora de goma (hule).

### Desventajas

La formación de la batería se produce lentamente, cada 72 horas, o a veces más. Se necesita más circuitos de carga. Se necesita más espacio de piso. Rendimiento eléctrico de la batería inferior, hasta en 10 – 15%. Alta temperatura de la batería durante las primeras 6 – 8 horas, difícil de controlar especialmente durante el verano, lo cual hace más lenta la formación y reduce la efectividad del expansor en las placas negativas.

#### **2.2.4 PRUEBA DE ALTA INTENSIDAD DE DESCARGA A LA BATERIA.**

El automóvil actual está diseñado de forma tal que la batería sirve únicamente para un propósito básico: arrancar el motor del auto. Todos los demás requerimientos eléctricos son, o extremadamente pequeños en demanda o funcionarían fácilmente con el alternador accionado por el motor mismo. Con lo anterior en mente, el fabricante de baterías deberá estar sustancialmente interesado en ver que cada una de sus baterías automotrices pueda llevar a cabo la función principal única de arrancar el motor. Para responder a este requerimiento, muchos fabricantes de baterías prueban cada una de las baterías a régimen alto, ya sea en el momento en que la batería húmeda está lista para enviarse al almacén o en el mismo momento en que la batería está siendo embarcada.

El equipo para este fin, por lo general es hecho por el mismo fabricante de baterías o por alguna compañía de construcción eléctrica del área local o adquirido con fabricantes de equipo para plantas de baterías en el mercado libre.

Como el arrancar un motor automotriz de combustión interna implica altos regímenes de descarga eléctrica por parte de la batería y (a veces) bajas temperaturas, la prueba establecida para esta función se le conoce como la "Prueba de Funcionamiento en Arranque", también conocida como la "Prueba de Arranque en Frío" y se define de la siguiente manera:

- El funcionamiento de Arranque es la cantidad en amperes de descarga que un acumulador puede proporcionar a 0°F (-17.8°C) durante 30 segundos, manteniendo un voltaje de 1.2 voltios o más por celda.
- En la Prueba de Control de Calidad que se lleva a cabo dentro de la planta, las baterías deben hacerse funcionar a regímenes más altos y por periodos de tiempo más corto por cada batería producida.

- Para baterías de 12 voltios, los regímenes de descarga son de aproximadamente 700 amperes mínimo. Como ejemplo, una batería normal de 11 placas, de 12 voltios, con una capacidad de 50 amperes - horas (capacidad nominal de 20 horas) deberá proporcionar 800 amperes durante 3 – 5 segundos.
- Use una terminación de descarga automática a régimen alto, una vez que se haya iniciado manualmente. La duración de la descarga deberá ser de 3 a 5 segundos.
- Las conexiones que van de la unidad de descarga a la batería de prueba deben hacerse rápidamente y desconectarse tan rápido y de forma tal que no se cause daño alguno a los bornes terminales de la batería.

Para evaluar las características de la prueba, se usa ya sea un amperímetro, para obtener la lectura máxima, o un voltímetro que permita ver el voltaje de la batería durante la prueba de descarga. Los instrumentos digitales (amperímetro o voltímetro) son de ayuda considerable para el operario en la prueba de alta intensidad.

Como los regímenes de carga en amperes varia de acuerdo con le número y tamaño de las placas en cada batería, los valores mínimos deben determinarse en base a la experiencia.

Los defectos en la batería que pueden detectarse con un probador de alta intensidad son:

- Conectores rotos.
- Postes conectores rotos.
- Soldadura de plomo defectuosa en conectores, bornes terminales o postes conectores.
- Separadores dañados o faltantes
- Salientes de placas rotos
- Celdas con corto.

El uso del probador de alta intensidad es de suma importancia, especialmente para un aplanta de baterías que apenas inicia su producción con operarios de

fabricación que no están todavía calificados para ejecutar su trabajo con pericia. La prueba permitirá detectar muchos defectos ocultos que de otra forma no podrían localizarse.

También es de suma importancia que los regímenes de descarga sean muy altos de modo que pueda detectarse cualquier defecto serio. Se ha demostrado que el uso de regímenes de descarga considerablemente más bajo, permiten pasar por alto defectos que sólo se manifestarán posteriormente.

# Capítulo 3

## 3. SOFTWARE “SISTEMA DE MONITOREO”

### 3.1 DESCRIPCION DEL SOFTWARE



CIB-ESPOL

*Sistema de Monitoreo*

Figura 3.1.- Portada de presentación del software Sistema de monitoreo

En la figura 3.1 podemos ver la pantalla de presentación del software. El software fue elaborado para monitorear el proceso de carga que se realiza en la fábrica Baterías Lux con una capacidad de monitoreo de 200 líneas de baterías, las líneas de carga están distribuidas en los diferentes cargadores, el número de baterías por línea depende de la capacidad del cargador por ejemplo un

cargador marca Firing Circuits de 300V, 30 A puede albergar de 15 a 20 baterías, este número varía de acuerdo al tamaño de la batería y al número de celdas de la batería así como a las dimensiones del espacio donde se cargan, en un banco o mesa.

El banco de baterías es donde se cargan las baterías. En el cuarto de Carga de baterías Lux existen bancos de baterías con 8 líneas cada uno, en dicho cuarto hay 25 Bancos de baterías.

En dicho software se monitorea 3 variables (Corriente, Temperatura, Voltaje) y calcula con ellas 3 parámetros (Horas de Carga, Amperios-Horas, Energía) y consta de las siguientes características:

- Una pantalla donde se dibuja en 3D el banco de baterías con 8 líneas, cada línea cambia de color de acuerdo al estado en el que se encuentra, por lo que existe un código de color.

- Gráfico de carga de cada línea, se grafica Corriente, Temperatura, Voltaje, Energía, Amperios Horas.
- Cuadro estadístico que calcula el número de baterías cargadas por mes y por cada tipo de batería en un año y se lo muestra en un diagrama de barras.
- Tabla de Alarmas, donde se manejan 4 alarmas con un color característico para cada una: Línea Cargada, Línea en circuito abierto, Alta Temperatura, Línea no programada.
- Tabla de datos que realiza consultas del proceso de carga vía SQL a una base de datos en Access.
- Una pantalla de Programación y Régimen de Carga en la cual se programa el tipo de batería y los Amperios-Horas de la línea de baterías a cargar según el régimen de carga que maneja el fabricante.
- Una pantalla de Configuraciones que maneja los valores que configuran el programa y la comunicación con las tarjetas de adquisición de datos.
- Una pantalla de calibración de los datos que se muestran en pantalla con una tabla de datos calibrados.

### 3.2 PANTALLAS DEL PROGRAMA Y SU FUNCIONAMIENTO.

#### *Menú del Programa*

En todas las pantallas del software existe una barra de menú que se puede observar en la figura 3.2, donde se accede a las diferentes pantallas del software. La barra consta de varios botones gráficos que identifican a cada pantalla. Al poner el cursor sobre los botones del menú, aparece un cuadro indicador de que pantalla corresponde ese botón.

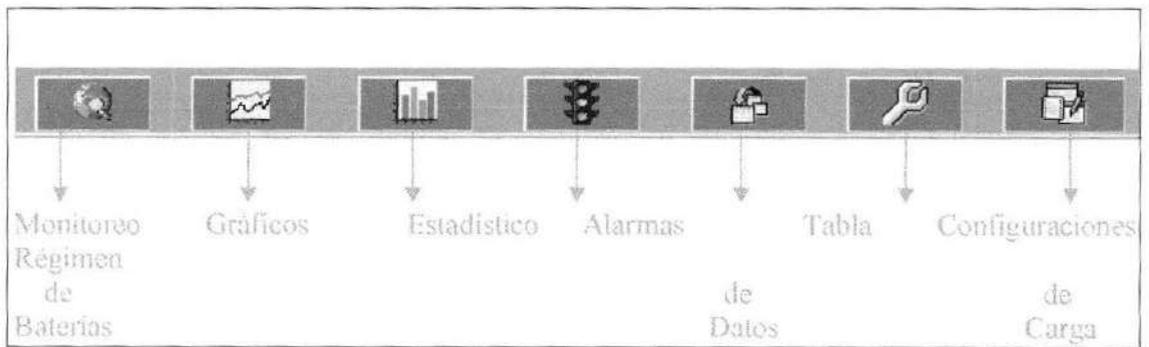


Figura 3.2.- Barra del Menú del programa.

### 3.2.1 PANTALLAS PRINCIPALES

#### *Monitoreo de Baterías*

En esta pantalla se puede observar de una manera dinámica el estado de carga de cada línea, el operador puede desplazarse a través de cada banco de baterías y por cada línea mediante las flechas del teclado y observará que cuando la batería está desconectada, es decir no circula corriente a través de ella, está de un color negro, cuando está cargando de un color azul y blanco que está cargada, además se puede hacer un monitoreo automático con desplazamientos del banco1 al 25 cada cierto tiempo y luego regresa al banco1, así el operador tiene visualización a través de todos los bancos y líneas y determinar el estado de cada línea de baterías, además si se produce alguna alarma en una línea un botón de Alarma aparece al lado de la línea de baterías y al hacer click en él se muestra la tabla de alarmas donde se puede determinar la fuente de la alarma. En la parte superior de la pantalla se indica el tipo de batería que está cargando en esa línea y el número de baterías determinado con la fórmula de la ecuación 3.1.

$$N^{\circ} = \left( \frac{\text{Voltaje\_Linea}}{\text{Voltaje\_Formación\_de\_Batería}} \right)$$

Ecuación 3.1.-

Donde N es el número de Baterías en la línea. Se calcula dividiendo el voltaje de línea monitoreado para el voltaje de formación de cada batería que es aproximadamente de 15 V, pero este cálculo se realiza a partir del 60% de la carga en el cual las placas positivas y negativas de la batería se están empezando a formar. En la figura 3.3 se muestra la pantalla de Monitoreo de Baterías.

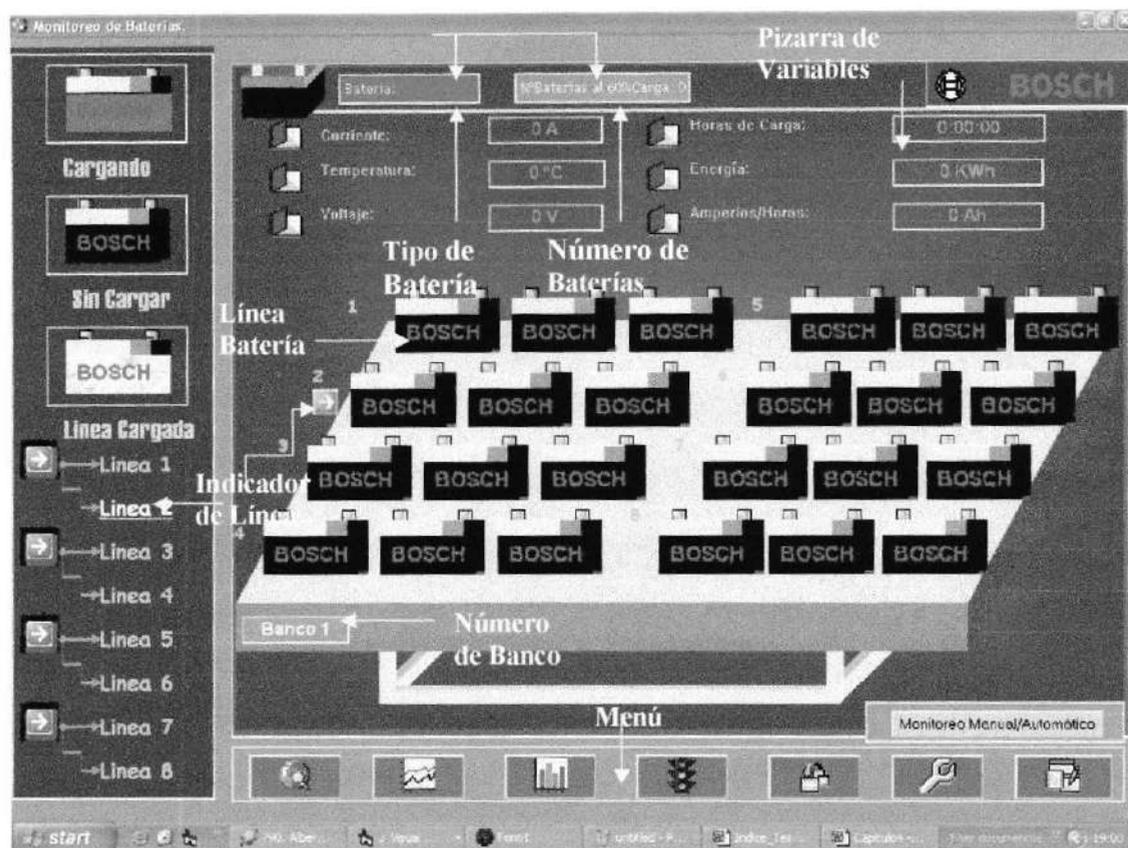


Figura 3.3.- Pantalla del Monitoreo de Baterías

### *Funcionamiento*



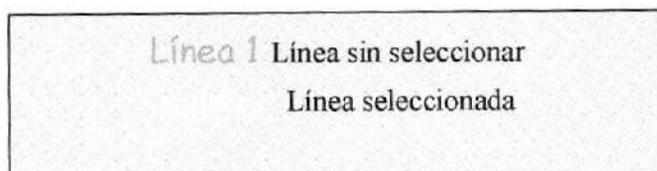
CIB-ESPOL

Para un buen funcionamiento, cada línea de baterías debe estar correctamente programada, sino no existirá una supervisión de líneas cargadas ni tampoco información guardada en la base de datos, el operador será el responsable de que este paso se ejecute para tener un buen monitoreo. En el caso de no

realizar la programación, el botón de alarma aparecerá al lado de la línea sin programar. Una vez las líneas estén programadas empezará el monitoreo.

Podemos realizar desplazamientos ascendentes a través de las líneas con la tecla KeyDown ↓ y desplazamientos descendentes con la tecla KeyUp ↑, así mismo desplazamientos ascendentes entre cada banco de baterías con la tecla KeyRight → y desplazamientos descendentes con la tecla KeyLeft ←, con los indicadores de líneas podremos saber en que línea nos encontramos. Existen dos tipos de indicadores:

- a) Dinámico es una flecha celeste que se mueve a través de las líneas según el movimiento de las flechas de teclado (Figura 3.3).
- b) Estático son indicadores que se encuentran en la parte inferior izquierda de la pantalla y se encuentran por grupos de 8 líneas, por ejemplo desde la Línea 1 a Línea 8 como se muestra en la figura 3.3, donde tiene un código de color para la línea que estamos seleccionando como se indica a continuación:



Cada vez que los indicadores están sobre una línea ocurren los siguientes eventos:

- En la parte superior donde se encuentra la "Pizarra de Variables" se muestra las variables monitoreadas corriente, temperatura, voltaje, horas de carga, energía, amperios-horas.
- En la parte superior donde se encuentran las etiquetas se muestra el tipo de batería que está cargando y en la otra etiqueta se muestra el número de baterías en dicha línea a partir del 60% de la carga.
- La etiqueta de número de banco de baterías, muestra el banco donde se encuentra la línea seleccionada.

Cuando fluye corriente a través de las líneas, estas se ponen de color azul, empieza el conteo de horas de carga, energía y amperios-horas. Cuando los amperios-horas alcanza el valor programado la batería se considera cargada y se pone de color Blanco pero continua con el conteo hasta que no detecte que la corriente sea cero, luego de la cual se encera todos los conteos, tipo de batería, número de baterías y la línea queda lista para otro ciclo de carga.

### *Gráficos*

Otra parte del monitoreo es por medio de gráficos, para esto se pulsa el botón de gráficos del menú donde aparece la pantalla de la Figura 3.4 que grafica cuatro variables a través del tiempo.

Pulsando el botón de selección, aparece una ventana donde se selecciona el tipo de batería de acuerdo al código que usa el fabricante para identificar sus modelos de baterías, hacemos clic en el modelo que deseamos graficar y aparecerá un cuadro de datos donde ingresamos el número de la línea y luego

el año del archivo, a continuación aparece el cuadro de archivos donde se escoge el archivo deseado.

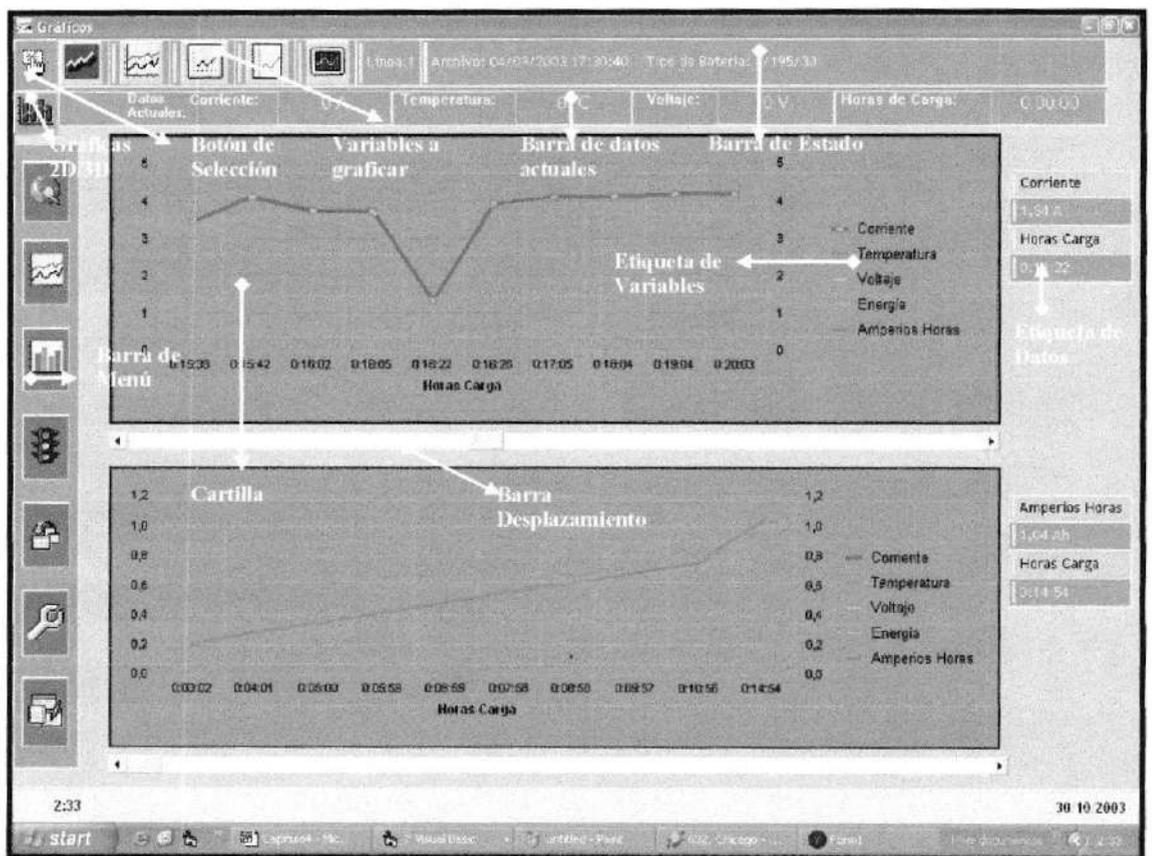


Figura 3.4.- Pantalla de Gráficos, grafica un ciclo de carga completa.

Una vez se escoge el tipo de Batería, Línea y Archivo podemos determinar que variable se van a graficar según la Figura 3.5.

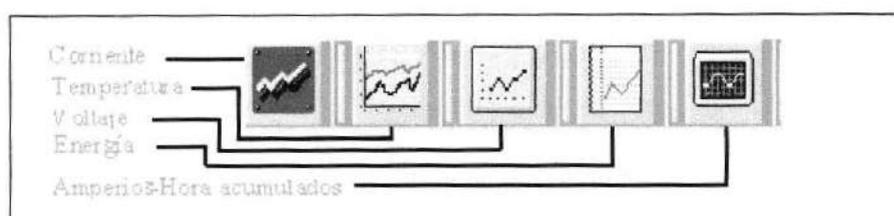


Figura 3.5.- Variables a graficar en la Barra de estado

Con la barra de desplazamiento de cada cartilla nos desplazamos a través del tiempo en todo el proceso de carga observando los cambios en la corriente de la línea, el voltaje, la energía consumida al final de la carga y la acumulación de amperios-horas.

### *Funciones*

El operador puede hacer uso de varias funciones en esta pantalla, a continuación las vamos ir mencionando:

- Gráficos 2D y 3D.
- Barra de desplazamiento, permite desplazarse a través de toda la carga, diez datos por cada desplazamiento.
- Movimiento y redimensionamiento del gráfico, arrastrándolo con el ratón de la PC.
- Selección de un punto o varios puntos en el gráfico, en caso de seleccionar un punto se mostrará en la etiquetas de datos.
- Actualización automática del gráfico cuando se selecciona una línea que se encuentra cargando.
- Gráfica la carga de la batería según el tipo de batería de la línea seleccionada
- División automática en el eje de la variable y división programada según el tiempo de monitoreo en el eje de las horas de carga.
- Una sola variable graficada por cartilla.

### *Partes de la Pantalla*

#### *a) Barra de Estado.*

Muestra el estado de los datos de la línea seleccionada, contiene varias etiquetas con datos de interés para el operador:

- Línea seleccionada.
- Archivo
- Tipo de Batería.

#### *b) Barra de Datos Actuales.*

Muestra los datos actuales de la línea seleccionada, esto sirve para que el operador no pierda la visualización de los datos que en ese momento están ocurriendo mientras revisa archivos anteriores.

#### *c) Barra de Menú.*

Podemos conectarnos entre pantallas con los botones del menú, cada uno indica gráficamente la pantalla que representa, pero además poniendo el cursor encima de cada botón se ve un cuadro que indica a que pantalla te dirige.



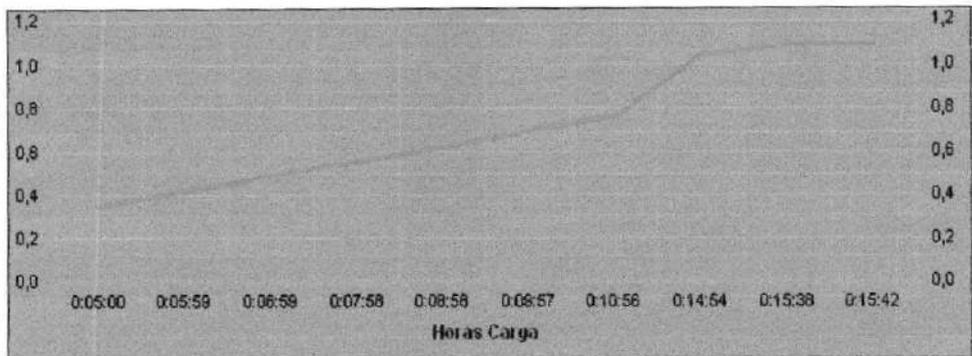
CIB-ESPOL

d) *Botón Selección.*

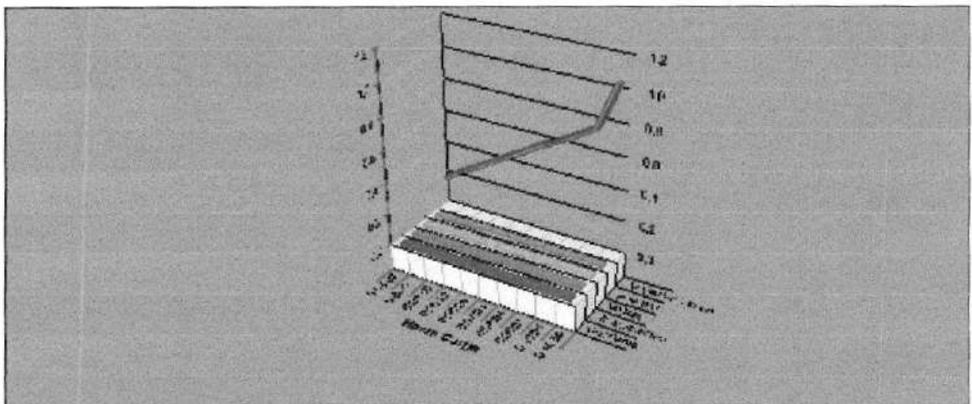
Se escoge el tipo de batería, línea y año del archivo.

e) *Botón Gráficos 2D/3D.*

En dos o tres dimensiones este botón nos permite observar los gráficos. En la Figura 3.6 se observa las dos formas de presentación de los gráficos.



a) Gráfico en 2D.



b) Gráfico en 3D.

Figura 3.6.- Forma de presentación de los Gráficos. En a) se encuentra graficado en 2 dimensiones y en b) en 3 dimensiones.

#### f) Etiquetas de Datos.

Se encuentran al lado derecho de la cartilla e indican los últimos datos de la cartilla cuando se mueve la barra de desplazamiento o muestra el dato de la

variable cuando se selecciona un punto en la cartilla. Muestra la variable graficada y las horas de carga.

*g) Etiquetas de Colores*

Indica el color con el que se grafica la variable:

**Rojo:** Corriente

**Verde:** Temperatura

**Amarillo:** Voltaje

**Naranja:** Energía

**Fucsia:** Amperios-Horas

*Estadístico*

Esta pantalla ofrece un cuadro estadístico que permite llevar un control de la producción de baterías por mes durante el año y de acuerdo al tipo de batería, se trata de un cuadro de barras como se puede ver en la figura 3.7.

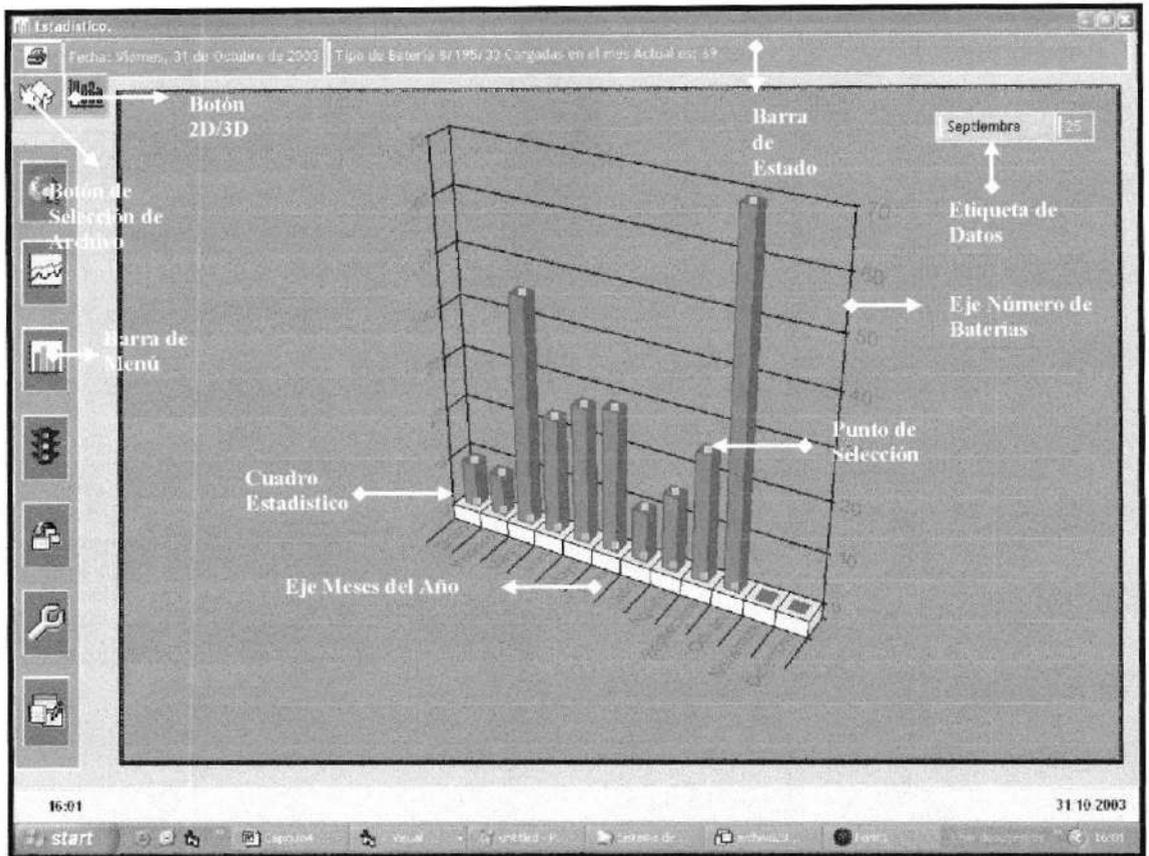


Figura 3.7.- Pantalla de Estadístico.

Para obtener un estadístico de la producción de baterías al mes se debe presionar el botón de selección de archivos con lo que aparece un cuadro donde se debe seleccionar el tipo de batería, luego aparece el Cuadro de Selección BOSCH donde ingresamos el año del archivo y luego aceptar, inmediatamente se dibuja el cuadro de barras e identifica la cantidad de baterías cargadas al mes.

### *Funciones*

Las funciones de esta pantalla se citan a continuación:

- Grafica el cuadro de barras en 2 y 3 dimensiones.
- Se puede escoger varios o un solo punto del cuadro pulsando en la parte superior de cualquiera de las barras e identificar el mes y la cantidad de baterías en la etiqueta de datos.
- Se puede redimensionar el gráfico primero se pulsa en el cuadro y luego de seleccionarlo se lo arrastra desde una de sus esquinas.

### *Partes de la pantalla*

#### *a) Barra de Estado.*

Muestra la fecha actual, el tipo de batería y el número de baterías cargadas al mes.

*b) Etiquetas de Datos...*

Muestra el número de baterías del cuadro de barras seleccionada con el ratón de la PC, conjuntamente con el mes al que corresponde.

*c) Botón Selección de Archivo.*

Se escoge el tipo de batería y el año del Estadístico.

*d) Botón Gráficos 2D/3D.*

Como en la pantalla de Gráficos, en dos o tres dimensiones este botón nos permite observar los gráficos del estadístico (Figura 3.8).

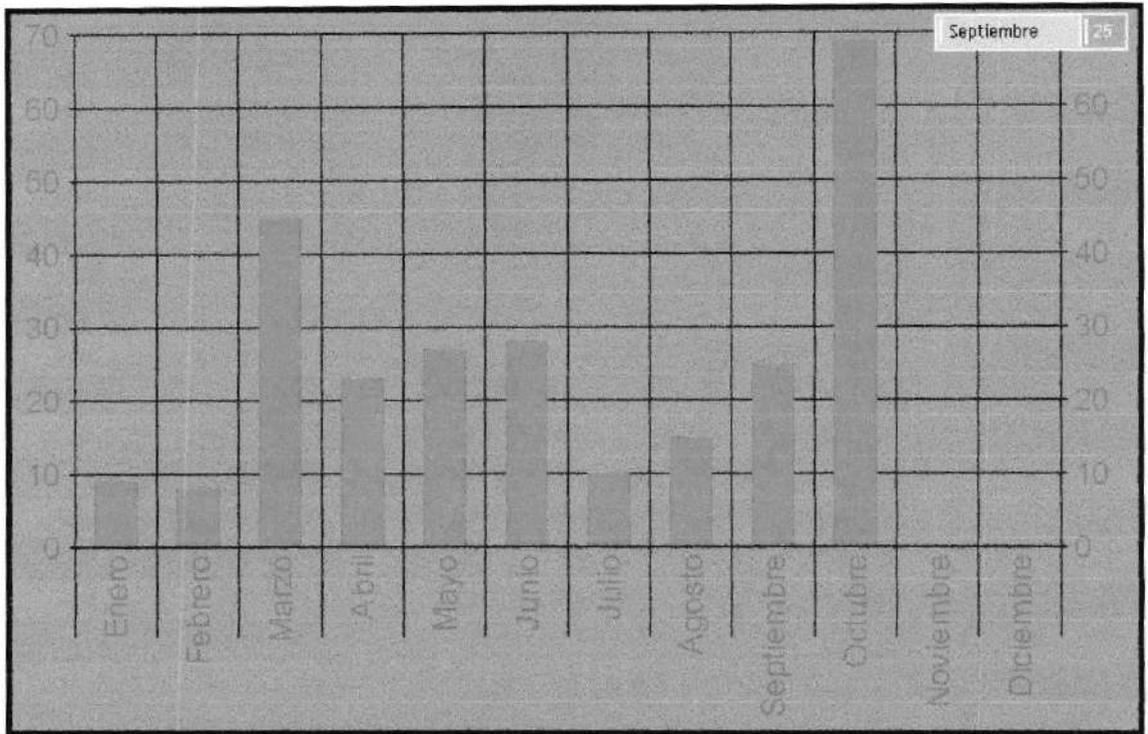


Figura 3.8.- Gráfico del estadístico de baterías cargadas por mes.

### 3.2.2 Pantallas de Tablas.-

En las tablas es donde se hacen consultas, se deben seguir casi los mismos pasos para consultas de gráficos y tablas de datos se pulsa el botón de archivo, aparece el cuadro de selección de tipo de batería, luego el cuadro de selección que pide el número de línea y finalmente el año del archivo. Para el cuadro estadístico nada más escogemos el tipo de batería del cuadro correspondiente y en el cuadro de selección, el año del archivo.

*Tabla de Datos.*

Con la tabla de datos de la figura 3.9 se pueden hacer consultas del proceso de carga de cada tipo de batería, los datos que presenta esta tabla son el tipo de batería, estado, fecha, horas de carga, corriente, voltaje, temperatura, energía, amperios-horas.

Tabla de Datos.

Fecha: Sábado, 01 de Noviembre de 2003 | Línea: 1 | Archivo: 04/09/2003 17:30:40 | Capacidad: 2,16Ah

Botón Selección de Archivo | Horas de Carga: 0:31:14 | Tipo de Batería: 8/195/33 | Barra de Estado

TipoBateria	Estado	Fecha	Horas de Carga	Volteje (V)	Corriente (A)	temperatura (°C)	Energía (KWh)	Capacidad (Ah)
8/195/33	Cargando	04/09/2003	0:03:02	0	4,2	32,7	0	0,21
8/195/33	Cargando	04/09/2003	0:04:01	0	4,2	32,7	0	0,28
8/195/33	Cargando	04/09/2003	0:05:00	0	4,2	32,7	0	0,35
8/195/33	Cargando	04/09/2003	0:05:59	0	4,2	32,7	0	0,42
8/195/33	Cargando	04/09/2003	0:06:58	0	4,2	32,7	0	0,49
8/195/33	Cargando	04/09/2003	0:07:58	0	4,2	32,7	0	0,56
8/195/33	Cargando	04/09/2003	0:08:58	0	4,2	32,7	0	0,62
8/195/33	Cargando	04/09/2003	0:09:57	0	4,2	32,7	0	0,69
8/195/33	Cargando	04/09/2003	0:10:56	0	4,2	32,7	0	0,76
8/195/33	Cargando	04/09/2003	0:14:54	0	4,2	32,7	0	1,04
8/195/33	Cargando	04/09/2003	0:15:38	0	3,45	32,7	0	1,09
8/195/33	Cargando	04/09/2003	0:15:42	0	4,12	32,7	0	1,09
8/195/33	Cargando	04/09/2003	0:16:02	0	3,74	32,7	0	1,12
8/195/33	Cargando	04/09/2003	0:16:05	0	3,74	32,7	0	1,12
8/195/33	Cargando	04/09/2003	0:16:22	0	1,34	32,7	0	1,14
8/195/33	Cargando	04/09/2003	0:16:26	0	3,93	32,7	0	1,14
8/195/33	Cargando	04/09/2003	0:17:05	0	4,11	32,2	0	1,18
8/195/33	Cargando	04/09/2003	0:18:04	0	4,12	32,2	0	1,25
8/195/33	Cargando	04/09/2003	0:19:04	0	4,19	32,7	0	1,32
8/195/33	Cargando	04/09/2003	0:20:03	0	4,19	32,7	0	1,39
8/195/33	Cargando	04/09/2003	0:21:02	0	4,19	32,7	0	1,46
8/195/33	Cargando	04/09/2003	0:22:01	0	4,19	32,2	0	1,53
8/195/33	Cargando	04/09/2003	0:23:00	0	4,19	32,7	0	1,6
8/195/33	Cargando	04/09/2003	0:23:59	0	4,19	32,2	0	1,66
8/195/33	Cargando	04/09/2003	0:24:59	0	4,19	32,7	0	1,74
8/195/33	Cargando	04/09/2003	0:27:57	0	4,19	32,2	0	1,94
8/195/33	Linea Car	04/09/2003	0:28:56	0	4,19	32,2	0	2,01
8/195/33	Linea Car	04/09/2003	0:29:55	0	4,19	32,2	0	2,08
8/195/33	Linea Car	04/09/2003	0:30:47	0	3,16	26,3	0	2,14
8/195/33	Linea Car	04/09/2003	0:30:53	0	2,11	32,2	0	2,14
8/195/33	Linea Car	04/09/2003	0:30:54	0	2,11	32,2	0	2,15
8/195/33	Linea Car	04/09/2003	0:31:13	0	0,1	23,9	0	2,16
8/195/33	Linea Car	04/09/2003	0:31:14	0	0	32,7	0	2,16
8/195/33	Fin de Car	04/09/2003		0	0	0	0	0

Figura 3.9.- Tabla de Datos



CIB-ESPOL

Para hacer una consulta debemos presionar el botón de selección de archivo, luego de lo cual tenemos una ventana para escoger el tipo de batería (figura 3.10 literal a), a continuación aparece el cuadro de selección BOSCH (figura

3.10 literal b y c) donde nos pide ingresar el número de línea y el año del archivo, el programa procede a realizar la consulta donde aparecen todos los archivos existentes, se procede a pulsar en la casilla del archivo que queremos consultar y luego tenemos la tabla de la figura 3.9.

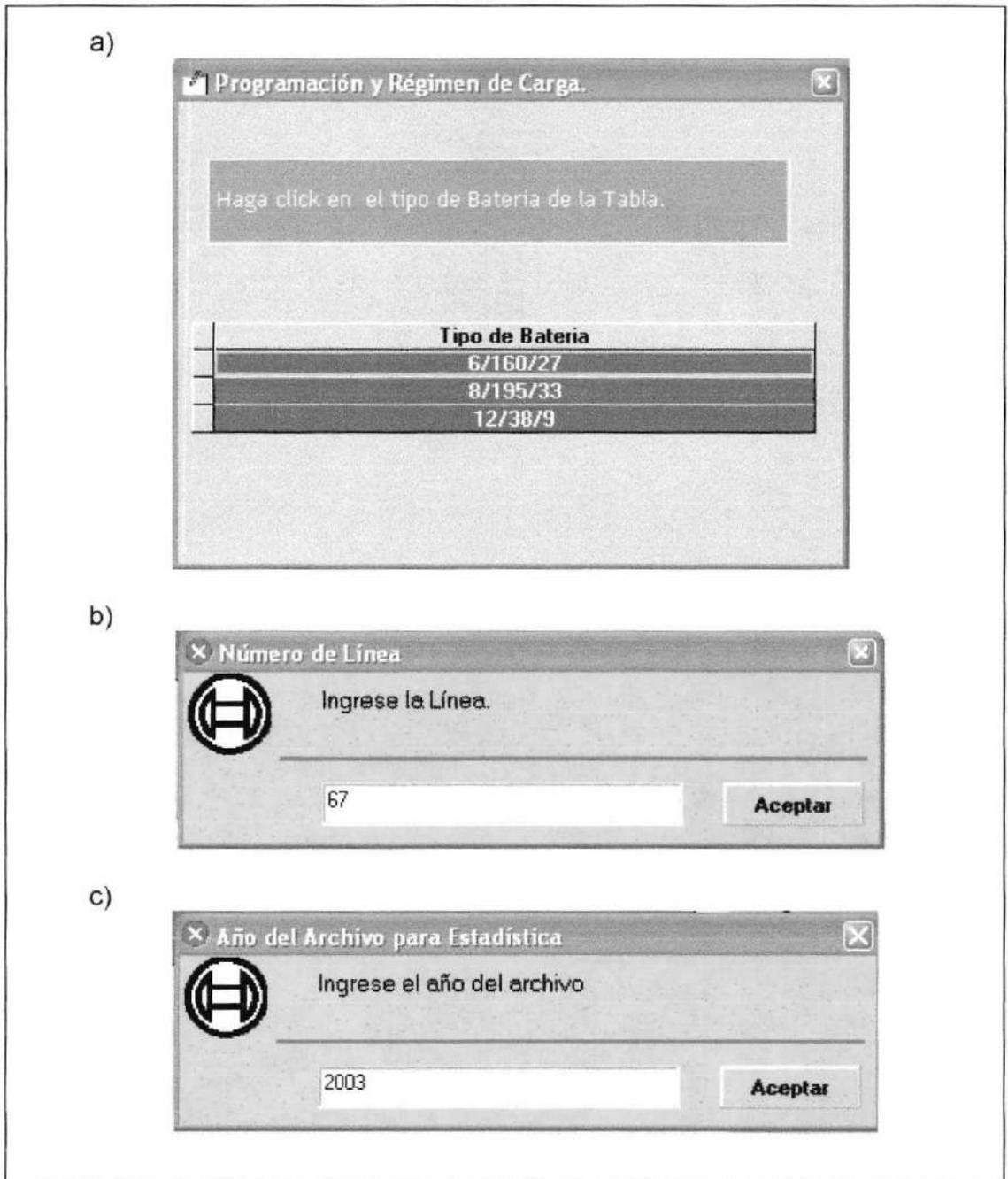


Figura 3.10.- a) Cuadro para seleccionar el tipo de batería, b) Cuadro de Selección BOSCH, pide el número de línea, c) Cuadro de Selección BOSCH, pide el año del archivo.

### *Formato del Archivo*

Para hacer las consultas en las tablas y para los gráficos se necesita un archivo que identifica el proceso de carga que necesitamos consultar. El formato del archivo consta de la fecha y la hora en que se programó la carga de determinado tipo de batería.

Archivo
04/09/2003 17:30:40
04/09/2003 18:46:04
04/09/2003 18:48:33
04/09/2003 19:59:10
04/09/2003 21:13:35
05/09/2003 0:12:49
05/09/2003 15:09:18
05/09/2003 15:25:23
09/09/2003 14:44:42
09/09/2003 14:51:44
09/09/2003 15:15:35

Figura 3.11.- Tabla de archivos, son todos los archivos de Carga de un cierto tipo de batería

### *Funciones*

- Consultas de carga de baterías de acuerdo al tipo o modelo.
- Redimensionamiento de la barra de estado y de las celdas de la tabla con funciones parecidas a las tablas de Excel.
- No permite el ingreso de datos en las tablas.
- Actualización manual de las consultas.

### *Partes de la pantalla*

#### *a) Barras de Estado.*

En la barra de estado se encuentra información sobre la consulta realizada como fecha, línea, archivo, capacidad de amperios horas al final de la carga, horas de carga finales, tipo de baterías.

#### *b) Botón Selección de Archivo.*

Con este botón podemos seleccionar el archivo que queremos consultar.

*c) Botón Actualización.*

Actualiza la tabla, sobretodo si la consulta es de una línea de baterías que se encuentra durante un proceso de carga.

*d) Tabla de Consulta.*

Es la tabla donde se sitúan los datos de la consulta, es una tabla semejante y con funciones parecidas a las tablas de Excel, la diferencia es que la tabla de consulta no permite ingresar datos en ella. Sus celdas son reajustables pulsando en cada división de la barra de estado y además pueden intercambiar de posición cada celda de la barra de estado solamente arrastrando cada celda a la posición que se desee.

*Programación y Régimen de Carga*



Esta pantalla consta de una tabla en la cual se encuentra el régimen de carga del fabricante, dicho régimen son los datos de fabricación de la batería, como el amperaje que deben fijar en los cargadores para su formación, como también las horas de carga.

Como podemos ver en la figura 3.12 el régimen de carga trae el tipo de batería que fabrican, número de placas positivas y negativas, cantidad de amperios-horas, amperaje de carga y horas de carga.

**Programación y Régimen de Carga.**

Línea: 13

Tipo de Batería: 8/195/33

Ah: 0.1

Horas: 40

Nuevo ciclo de Carga para la línea en la lista.

Nuevo ciclo de carga para todas las líneas.

Ah actuales: [ ]

Id	Tipo de Bate	N° Placa +	N° Placa -	Cantidad de Ah	Operaje de Carga	Horas de Carga
1	6/160/27	14	13	560	1	40
2	8/195/33	17	16	0.1	7	40
3	12/38/9	6	4	152	5	30

Barra de Programación

Etiquetas

Tabla Régimen de Carga

Lista Desplegable

Botón y Cuadro de Texto Ah Actuales

Tipo de Batería Programada

Botones Nuevo ciclo de Carga

Figura 3.12.- Tabla Programación y régimen de Carga.

Esta instancia es muy importante porque es aquí donde se programa la línea a cargar, el operador debe primero escoger la línea de la lista desplegable de la barra de programación luego según el tipo de batería a cargar buscamos su correspondiente Id en la tabla régimen de carga que se encuentra en la primera columna y lo ponemos en el cuadro de texto al lado de la etiqueta que dice "Tipo de Batería", automáticamente aparecerán en las etiquetas de datos las horas de carga y los amperios-horas de el tipo de batería a cargar.

### *Funciones*

- Programación de las líneas de batería, se pueden programar 200 líneas de baterías.
- Encerar los datos de programación de una línea o de todas las líneas en casos en que la línea sea mal programada.
- En ocasiones el conteo de amperios horas del sistema puede perderse o puede haber un desfase con el conteo real debido a un corte de energía en donde la PC puede apagarse o en casos en que la línea lleve algunas

horas de carga y no hemos programado la línea en tal caso podemos actualizar el conteo.

- La tabla de régimen de carga soporta actualizaciones en la base de datos del sistema.

### *Partes de la Pantalla*



#### *a) Barra de programación*

Es la barra donde se encuentra la lista desplegable, cuadro de texto y etiquetas.

#### *b) Lista Desplegable*

Es parecido a un cuadro de texto pero la diferencia es que al pulsar en él sale una lista que se despliega hacia abajo mostrando las líneas de 1 a 200.

#### *c) Etiquetas.*

Se encuentran en la barra de programación e indican el tipo de batería, las horas de carga y amperios-horas programados.

*d) Cuadro de texto.*

Hay dos cuadros de texto uno está en la barra de programación y es para ingresar el Id (Identificación del tipo de batería) y el otro es para ingresar los amperios horas actuales de la línea que se encuentra seleccionada en la lista desplegable.

*e) Botón de Amperios Horas actuales.*

Una vez ingresado el valor de amperios horas actuales se pulsa en este botón para actualizar los amperios horas de la línea en la lista desplegable.

*f) Botón de Nuevos ciclos de Carga.*

Son dos botones que son para encerrar las horas de carga y conteo de amperios horas así como los demás datos de programación para iniciar un ciclo de carga.

*g) Tabla Régimen de Carga.*

Esta tabla contiene datos del régimen de carga del fabricante para cargar sus baterías.

### Calibración

Con la pantalla de la figura 3.13 podemos calibrar vía software los datos que llegan de las tarjetas de adquisición de datos y que se muestran en el sistema, para cada variable hay una forma de calibración. Para iniciar una calibración pulsamos el botón Calibrar, aparece los controles de calibración y luego se procede a realizar la calibración deseada.

The screenshot shows a software window titled 'Calibración' with a sub-header 'TABLA DE CALIBRACION'. On the left, there is a vertical bar with a wrench icon and the word 'CALIBRAR'. The main area contains a table of calibration parameters for five variables:

Variable	Value	Unit
mv max. en Shunt	0,3	
Corriente max. Cargador	30	
Corriente Medida	0	AMPS
Tempertatura Medida	1	°C
Voltaje Medido	1	VOLTS

In the bottom right corner, there is a circular seal of the 'ESCUELA SUPERIOR POLICIA' and the text 'CIB-ESPOL'.

Figura 3.13. - Cuadro de Calibraciones de las variables monitoreadas.

### *Calibración de corriente al 100%*

Cuando una tarjeta de adquisición se conecta al sistema se debe hacer una primera calibración de corriente al 100%, esto significa que se debe realizar con la corriente máxima en la línea.

Los pasos son los siguientes:

- 1.- Escoger la línea a calibrar en la lista desplegable.
- 2.- Fijar la corriente máxima en el cargador que contiene esa línea.
- 3.- Medir con un Voltímetro el Shunt de esa línea e ingresarlo en el cuadro de texto que corresponde a "mv max. en Shunt".
- 4.- Medir con un amperímetro de gancho la corriente máxima en la línea e ingresar el valor medido en el cuadro de texto que corresponde a "Corriente max. Cargador".
- 5.- Terminar la calibración pulsando el botón de Calibración de Corriente "AMPS".

Las relaciones para determinar la corriente que leen las tarjetas de adquisición de datos se muestran según la ecuación 3.2:

$$I[A] = \frac{V(\text{Voltaje\_leído})}{A(\text{Ganancia\_Amplificación})} \times \frac{I_{max}}{mV\_Shunt}$$

Ecuación 3.2

#### *Calibración de corriente Durante la Carga*

- 1.- Escoger la línea en la lista desplegable.
- 2.- Medir con amperímetro de gancho el valor de la corriente en la línea correspondiente e ingresarlo en el cuadro de texto correspondiente a "Corriente Medida"
- 3.- Terminar la calibración pulsando el botón de Calibración de Corriente "AMPS".

La corriente calibrada de la ecuación 3.3 se determina restando:

$$\begin{aligned} \text{Factor\_Calibración} &= I[A] - I[\text{Medida}] \\ I[\text{Calibrada}] &= I[A] \pm \text{Factor\_Calibración} \end{aligned}$$

Ecuación 3.3

#### *Calibración Voltaje y Temperatura*

El voltaje de línea de la mayoría de los cargadores se encuentra alrededor de los 300V DC, por lo que la relación que se usa es la del divisor de voltaje para calcularlo:

$$V_{\text{Línea}} = V(\text{Voltaje leído}) \times \frac{6856}{56}$$

Ecuación 3.4

La relación de la ecuación 3.5 es de la temperatura, pero en el capítulo 5 se la analiza con más detalle:

$$T[{}^{\circ}C] = V[\text{Voltaje leído}] \times \frac{70}{5}$$

Ecuación 3.5

La relación indica que a 70°C equivalen 5 Voltios en la lectura de la adquisición de datos. Los Pasos para la calibración se enumeran a continuación:

#### *Calibración vía Software*

- 1.- Escoger la línea en la lista desplegable.
- 2.- Medir con un voltímetro el voltaje de línea e ingresarlo en el cuadro de texto correspondiente a "Voltaje Medido".
- 3.- Medir con un termómetro la temperatura medida e ingresarlo en el cuadro de texto correspondiente a "Temperatura Medida".
- 4.- Terminar la calibración pulsando el botón de Calibración de Voltaje "VOLTS" y el botón de Calibración de Temperatura "°C".

Seguidamente a la calibración, se muestran las relaciones de calibración del voltaje y de la temperatura finalmente llegan a la ecuación 3.6:

$$Factor\_Calibración = VLinea - V[Medido]$$

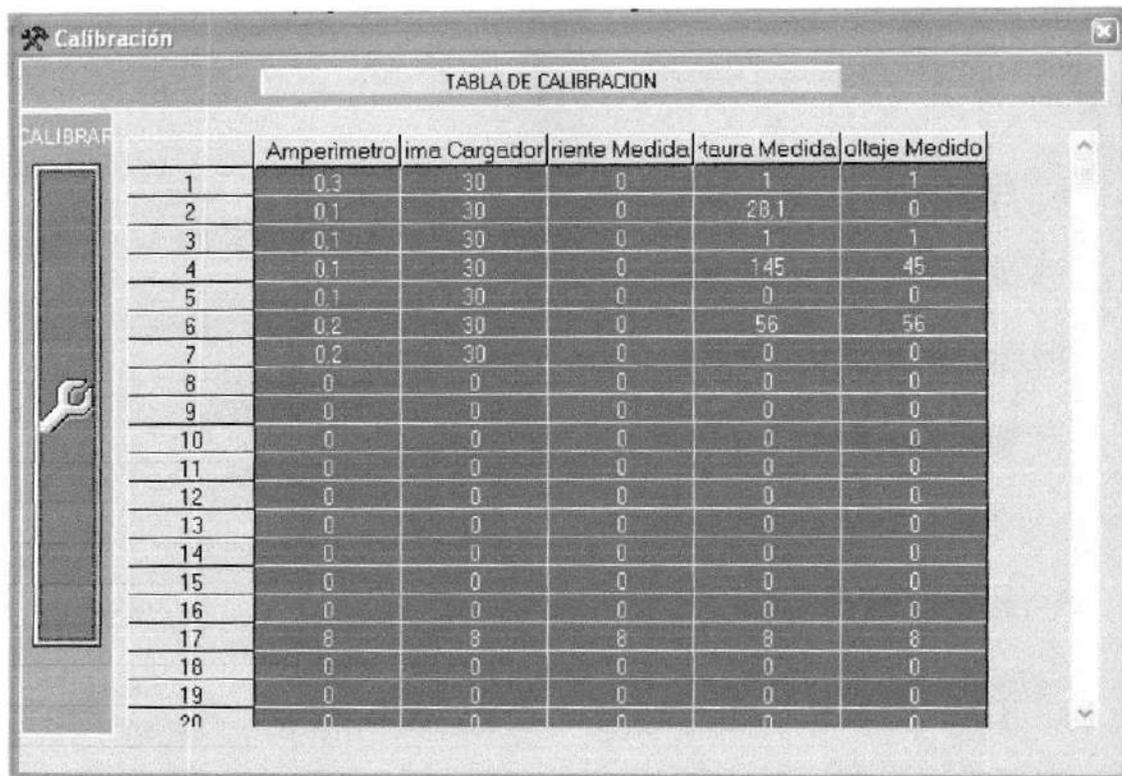
$$V[Calibrado] = VLinea \pm Factor\_Calibración$$

$$Factor\_Calibración = T[^{\circ}C] - T[Medida]$$

$$T[Calibrada] = T[^{\circ}C] \pm Factor\_Calibración$$

Ecuación 3.6

Para comprobar las calibraciones realizadas se pulsa en el botón de Calibración y luego aparece una tabla como en la figura 3.14 donde aparecen todas las calibraciones de todas las líneas.



The screenshot shows a software window titled "Calibración" with a sub-header "TABLA DE CALIBRACION". On the left side, there is a vertical toolbar with a wrench icon and the label "CALIBRAR". The main area contains a table with 20 rows and 6 columns. The columns are labeled: "Linea", "Amperimetro", "Corriente Cargador", "Corriente Medida", "Temperatura Medida", and "Voltaje Medido". The data in the table is as follows:

Linea	Amperimetro	Corriente Cargador	Corriente Medida	Temperatura Medida	Voltaje Medido
1	0.3	30	0	1	1
2	0.1	30	0	20.1	0
3	0.1	30	0	1	1
4	0.1	30	0	145	45
5	0.1	30	0	0	0
6	0.2	30	0	56	56
7	0.2	30	0	0	0
8	0	0	0	0	0
9	0	0	0	0	0
10	0	0	0	0	0
11	0	0	0	0	0
12	0	0	0	0	0
13	0	0	0	0	0
14	0	0	0	0	0
15	0	0	0	0	0
16	0	0	0	0	0
17	8	8	8	8	8
18	0	0	0	0	0
19	0	0	0	0	0
20	0	0	0	0	0

Figura 3.14.- Tabla de Calibraciones donde se pueden observar los datos calibrados, la primera columna indica el número de línea, la segunda los milivoltios en el shunt, la tercera la corriente medida en un instrumento de medición, la cuarta y quinta temperatura y voltaje medidos.

#### *Tabla de Alarmas*

La Tabla de Alarmas en la barra de menú se simboliza con un semáforo como el de la figura 3.15, es un botón de fondo azul, cuando se produce alguna alarma independientemente de la pantalla en la que estemos el botón de Alarma empieza a parpadear de azul a rojo, indicando que se ha producido una alarma

y que debemos pulsar el botón para determinar a cual corresponde. El botón de alarma se puede apreciar en la figura 3.16.

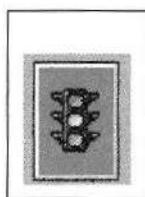


Figura 3.15.- Botón de Alarma

La tabla de Alarmas nos indica todas las fuentes de Alarmas en todas las líneas, además cada fuente de alarma tiene un código de color que la identifica, estas son:

- Rojo** = Circuito Abierto
- Azul** = Alta Temperatura
- Verde** = No Programada
- Naranja** = Línea Cargada

Tabla de Alarmas	
11:07:30	No hay corriente en la Línea 1 ,está desconectada.
11:07:30	La Línea 1 tiene alta temperatura.
11:07:30	No hay corriente en la Línea 2 ,está desconectada.
11:07:30	La Línea 2 tiene alta temperatura.
11:07:30	No hay corriente en la Línea 3 ,está desconectada.
11:07:30	La Línea 3 tiene alta temperatura.
11:07:30	No hay corriente en la Línea 4 ,está desconectada.
11:07:30	La Línea 4 tiene alta temperatura.
11:07:30	No hay corriente en la Línea 5 ,está desconectada.
11:07:30	La Línea 5 tiene alta temperatura.
11:07:30	No hay corriente en la Línea 6 ,está desconectada.
11:07:30	La Línea 6 tiene alta temperatura.
11:07:30	No hay corriente en la Línea 7 ,está desconectada.
11:07:30	La Línea 7 tiene alta temperatura.
11:07:30	No hay corriente en la Línea 8 ,está desconectada.

Figura 3.16.- Tabla de Alarmas

### *Función*

La función de la tabla es permitir al operador estar al tanto de lo que está pasando en el cuarto de carga directamente en su PC, a través de esta tabla.

### *Fuentes de Alarma*

#### *a) Circuito Abierto*

Es cuando por la línea de baterías no pasa corriente, aún cuando el cargador está programado para que cierta cantidad de corriente pase por la línea.

#### *b) Alta Temperatura*

Cuando la temperatura dentro de la batería en sus celdas, sube por arriba de los niveles programado por el operador.

#### *c) No Programada*

Cuando la línea que se está cargando no ha sido programada por el operador, por lo tanto el programa no sabrá cuando la línea termina su carga.

#### *d) Línea Cargada*

Cuando la línea ha cumplido una cantidad programable de amperios horas de carga, el programa lo detecta y lo indica en la tabla de alarmas.

### *Partes de la Pantalla*

#### *a) Barra de Fuentes de Alarma.*

Se encuentra en la parte superior de la pantalla e indica las fuentes de alarma con su respectivo código de color.

#### *b) Tabla de Alarmas.*

La tabla de alarmas contiene en la primera columna la hora en la que ocurrió la alarma y en la segunda columna todas las fuentes de alarma para cada una de las líneas.

### **3.2.3 PANTALLAS DE CONFIGURACION.**

La pantalla configuraciones permite al operador establecer y configurar los diferentes elementos que componen el sistema de monitoreo.

Configuración tiene 5 elementos:

- a. Comunicación.
- b. Dispositivos.

c. Especificaciones.

d. Errores.

e. Contraseñas.

a.- Comunicación

La etiqueta comunicación en la pantalla de configuraciones de la Figura 3.17 se subdivide en:

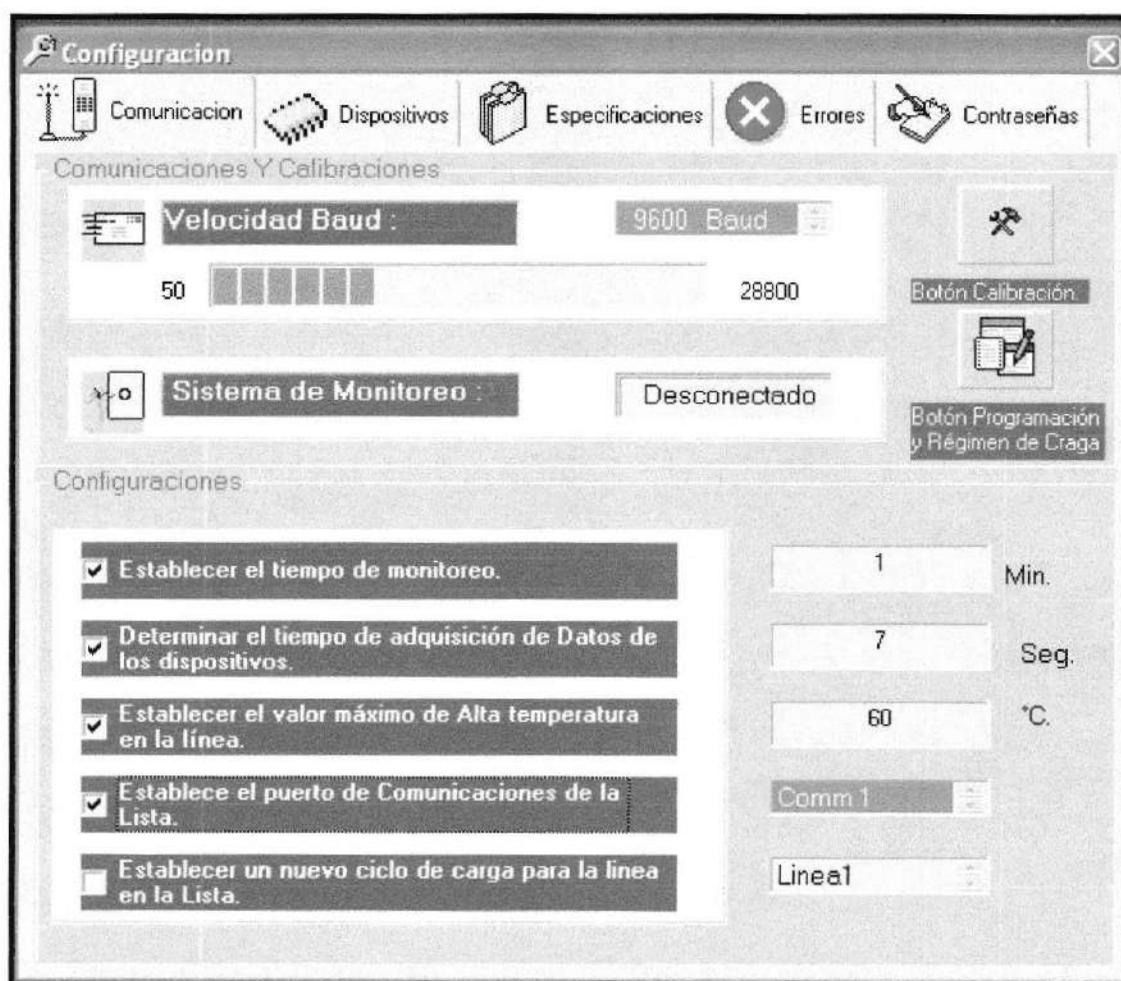


Figura 3.17.- Pantalla de Configuración, etiqueta de Comunicaciones.

### Comunicaciones y calibraciones



CIB-ESPOL

En la cual la velocidad es en Baud, de la comunicación entre los dispositivos de adquisición de datos y la PC, hay una lista desplegable donde podemos escoger

la velocidad de comunicación pero en esta versión, el software lo fija automáticamente en la norma más usada de 9600 Baud. Gráficamente se muestra con una barra de medición de velocidad que va desde 50 Baud a 28800 Baud.

En la parte inferior de comunicaciones y calibraciones se encuentra el estado de conexión del sistema, un cuadro de estado indica si el sistema está conectado a los dispositivos de adquisición de datos, gráficamente se muestra una toma de conexión en la parte inferior izquierda de comunicaciones y calibraciones.

En la parte derecha de comunicaciones y calibraciones está dos botones que te conectan con las pantallas de Calibración y Programación y Régimen de Carga vistas anteriormente.

### *Configuraciones*

En configuraciones tenemos unas casillas de verificación y a lado de cada uno cuadros de textos o listas desplegadas para introducir algún valor de configuración o seleccionarlos de las listas.

Los cuadros de verificación son los siguientes:

1.- Establecer el tiempo de monitoreo.

Cuadro de texto en el cual se ingresa el tiempo en el que cada vez el software guarda los datos del sistema y de adquisición de datos a una base de datos.

2.- Determinar el tiempo de adquisición de Datos de los dispositivos.

En el cuadro de texto correspondiente se ingresa el valor, que es el tiempo que le toma a los dispositivos en hacer un barrido, recogiendo todos los datos de los cargadores. Este tiempo debe ser racional para que los dispositivos recojan los datos que es de 6 a 7 segundos.

3.- Establecer el valor máximo de Alta temperatura en la línea.

Al subir la corriente en las líneas de baterías existe un aumento en la temperatura dentro de las celdas de la misma, por lo que hay un valor máximo de temperatura que depende del fabricante, este valor de temperatura se ingresa en el cuadro de texto correspondiente, en este caso es de 60 a 70°C.

4.- Establece el puerto de Comunicaciones de la Lista.

El ingreso de los datos de los dispositivos a la PC es por sus puertos de comunicación, dependiendo de la máquina en la que se instala el software, por

lo tanto hay tres puertos que se pueden seleccionar de una lista desplegable que son: Com1, Com2 y Com3, el operador debe escoger el correspondiente puerto disponible en su PC, por Default el puerto de comunicaciones es Com1.

5.- Establecer un nuevo ciclo de carga para la línea en la Lista.

Como ya se vio en la pantalla de Programación y Régimen de Carga, de una lista desplegable escogemos la línea a la cual queremos encerrar los datos que en ella se encuentran.

#### b.- Dispositivos

En la etiqueta Dispositivos hay un cuadro de texto que indica el número de Dispositivos conectados al sistema de monitoreo, esta información es actualizada por el sistema dependiendo del tiempo de adquisición de datos por ejemplo si lo fijamos en 7, cada 7 segundos veremos cuantos dispositivos hay conectados, por lo tanto sabremos si alguno está fallando con el número de dispositivos conectados y el número que nos informa el sistema. Existe una tabla que indica cada dispositivo conectado con su código y número de datos que envía, esta información es enviada por los dispositivos al sistema, y éste los ubica en la tabla. El operador debe entonces poner la ubicación del mismo, es decir en que cargador se encuentra y pulsando dos veces sobre el código del

dispositivo aparecerá una columna correspondiente a una lista desplegable para asignarle la línea que está monitoreando. En la parte inferior de la figura 3.18 tenemos dos casillas de verificación, que son para hacer algún cambio de líneas de baterías, y la segunda para iniciar el monitoreo.

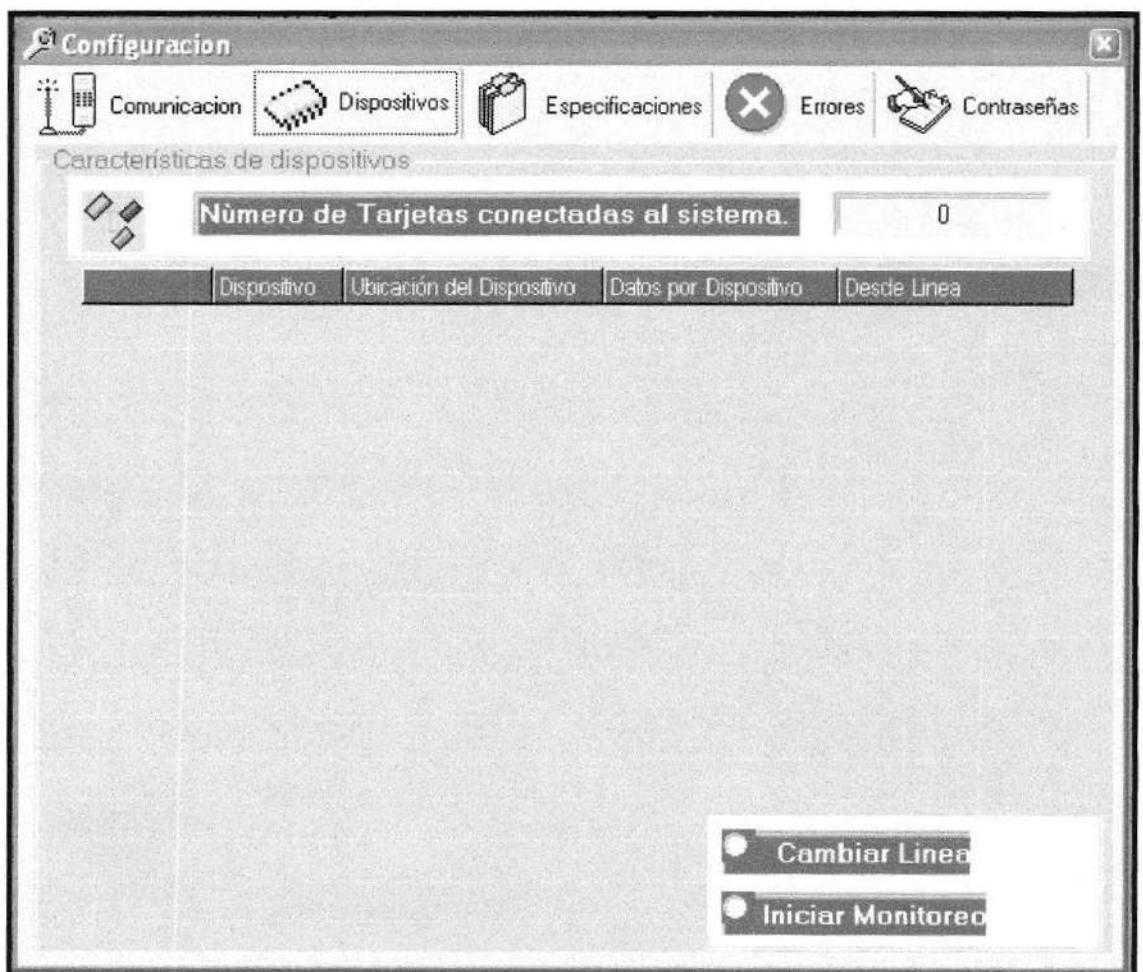


Figura 3.18.- Configuración, Etiqueta Dispositivos, contiene una tabla con Dispositivos, Ubicación del Dispositivo, Datos por Dispositivo, Desde Línea

### c.- Especificaciones

La etiqueta especificaciones de la figura 3.19, consta de un documento que indica el funcionamiento del sistema y de algunos parámetros que se deben considerar para su buen funcionamiento, además pasos para mantenimiento de los dispositivos, la base de datos del sistema, etc.

Lo especial de este documento es que hay un botón de actualizar que al pulsarlo actualiza el documento, dicho documento es un archivo en Word que se encuentra en la carpeta Sistema de Monitoreo que se crea al instalar el programa, la dirección de la carpeta es C:\Program Files\Sistema de Monitoreo, este puede ser un sistema de mantenimiento retroalimentado ya que se puede pedir una actualización a otra persona por ejemplo del departamento de mantenimiento, este lo revisa le hace los cambios necesarios y luego lo reenvía, todo esto vía Internet o una red local , el operador lo recibe en su correo y luego lo guarda en la dirección especificada, luego en el software pulsa actualizar y ya obtiene el documento con los datos solicitados.

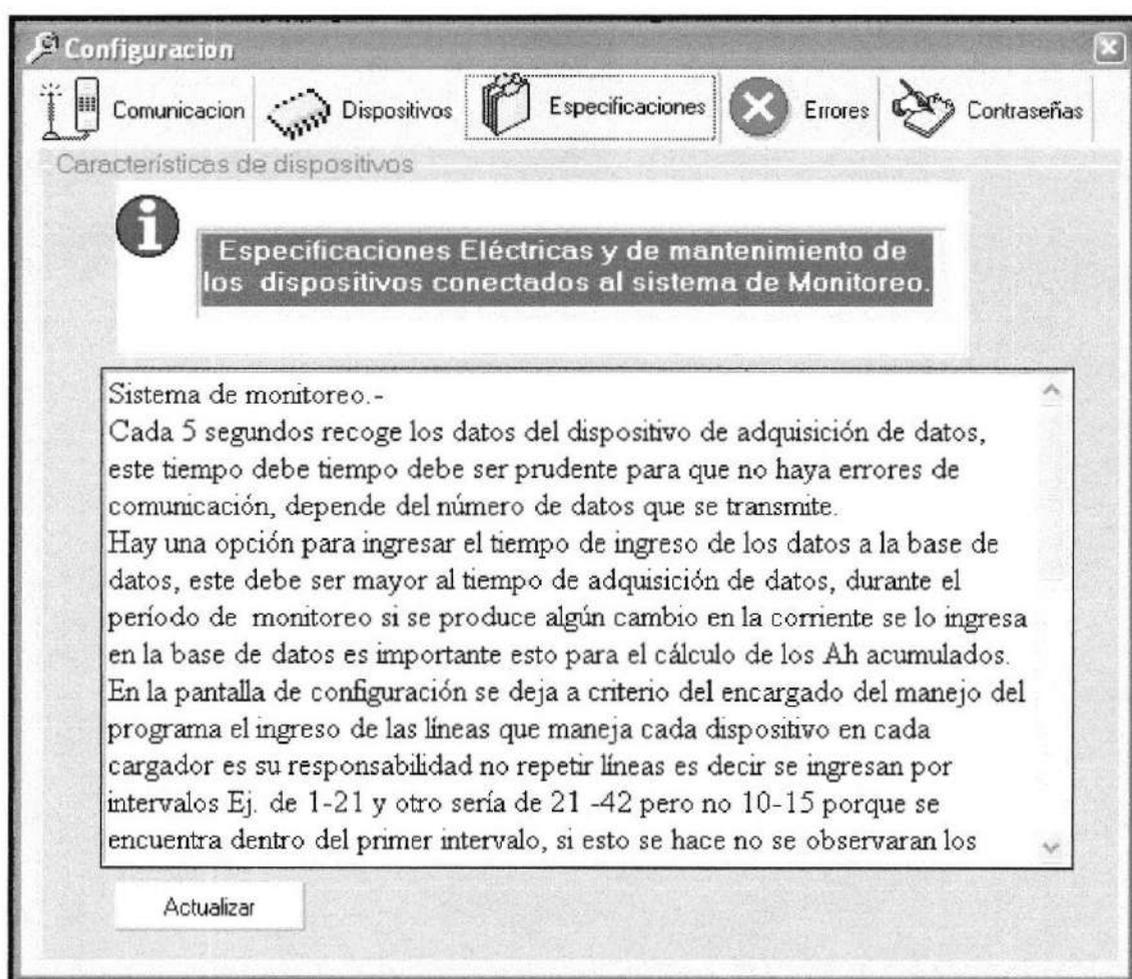


Figura3.19.- Configuración, etiqueta Especificaciones, las especificaciones eléctricas y de mantenimiento se encuentran en el cuadro central.

#### d.- Errores

Los errores en este caso son errores de comunicación entre los dispositivos y la PC, los cuales se muestran en un cuadro en la figura 3.120, el estado normal de funcionamiento es con el mensaje "Recibido los Datos", a continuación los enumeramos:

1. "Se ha recibido una interrupción."
2. "Error de trama"
3. "Datos perdidos."
4. "Desbordamiento del búfer de recepción de recepción."
5. "Error de paridad."
6. "Búfer de transmisión lleno."
7. "Error inesperado al recuperar DCB."
8. "Cambio en la línea CD."
9. "Cambio en la línea CTS."
10. "Cambio en la línea DSR."
11. "Cambio en el indicador de llamadas. "
12. "Recibido los Datos."
13. "Hay un SThreshold caracteres en el búfer de transmisión."

14. "Se ha encontrado un carácter EOF en la entrada."

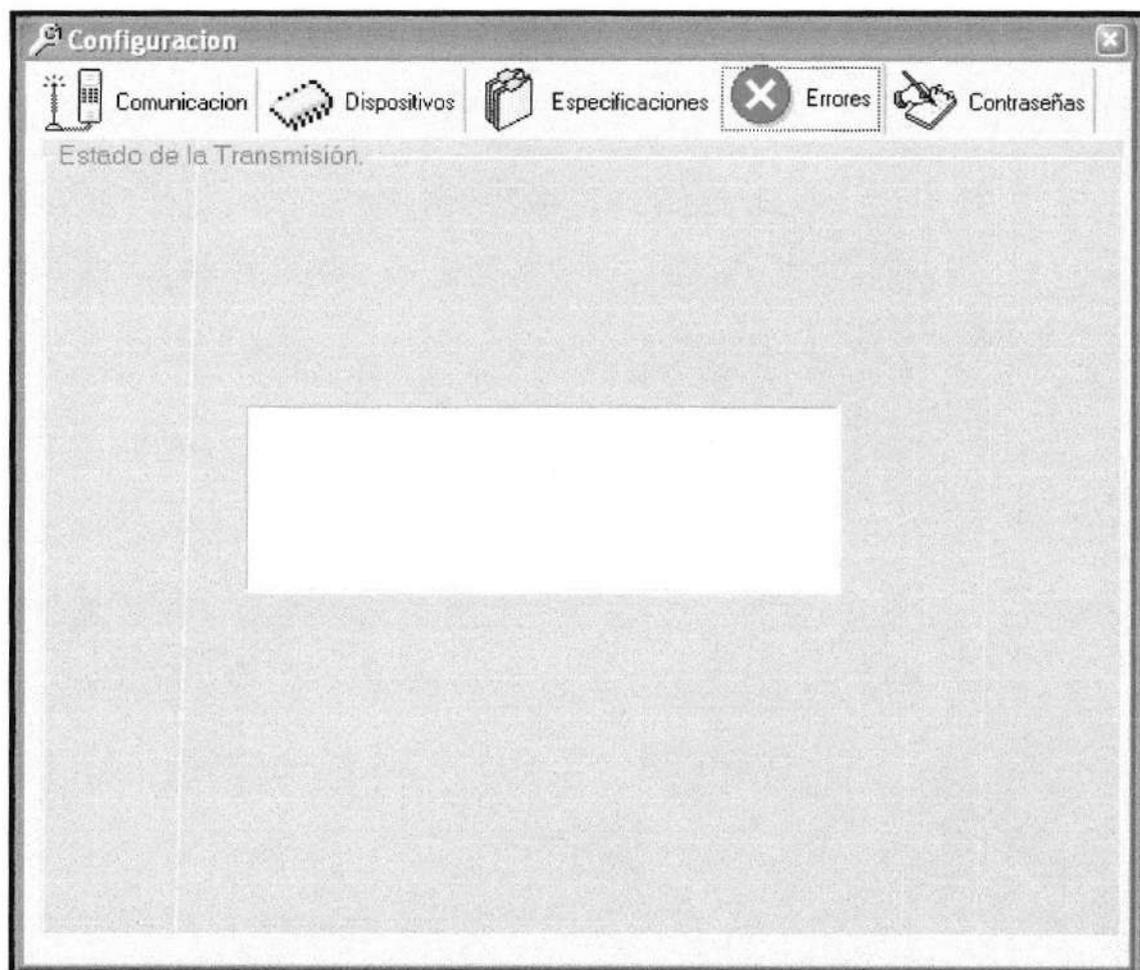


Figura 3.20.- Errores, etiqueta Estado de Transmisión, el estado de La transmisión de datos por el puerto de la PC se observa en el cuadro central.

#### e.- Contraseñas

En la etiqueta Contraseñas se guarda las contraseñas de Inicio de sesión, Cerrar Sesión y Contraseña de Configuración, cada clave tiene un nivel de seguridad que puede ser asignada a diferentes usuarios (Figura 3.21).

Para cambiar o ingresar una contraseña se ingresa la clave en el segundo cuadro de texto, luego se pulsa el botón de Cambiar y en el cuadro de arriba se mostrará la nueva contraseña, la clave de más alta seguridad es la contraseña de configuración ya que en Configuración, etiqueta contraseñas se encuentran las demás contraseñas, le siguen en orden de seguridad Cerrar sesión e Inicio de Sesión.

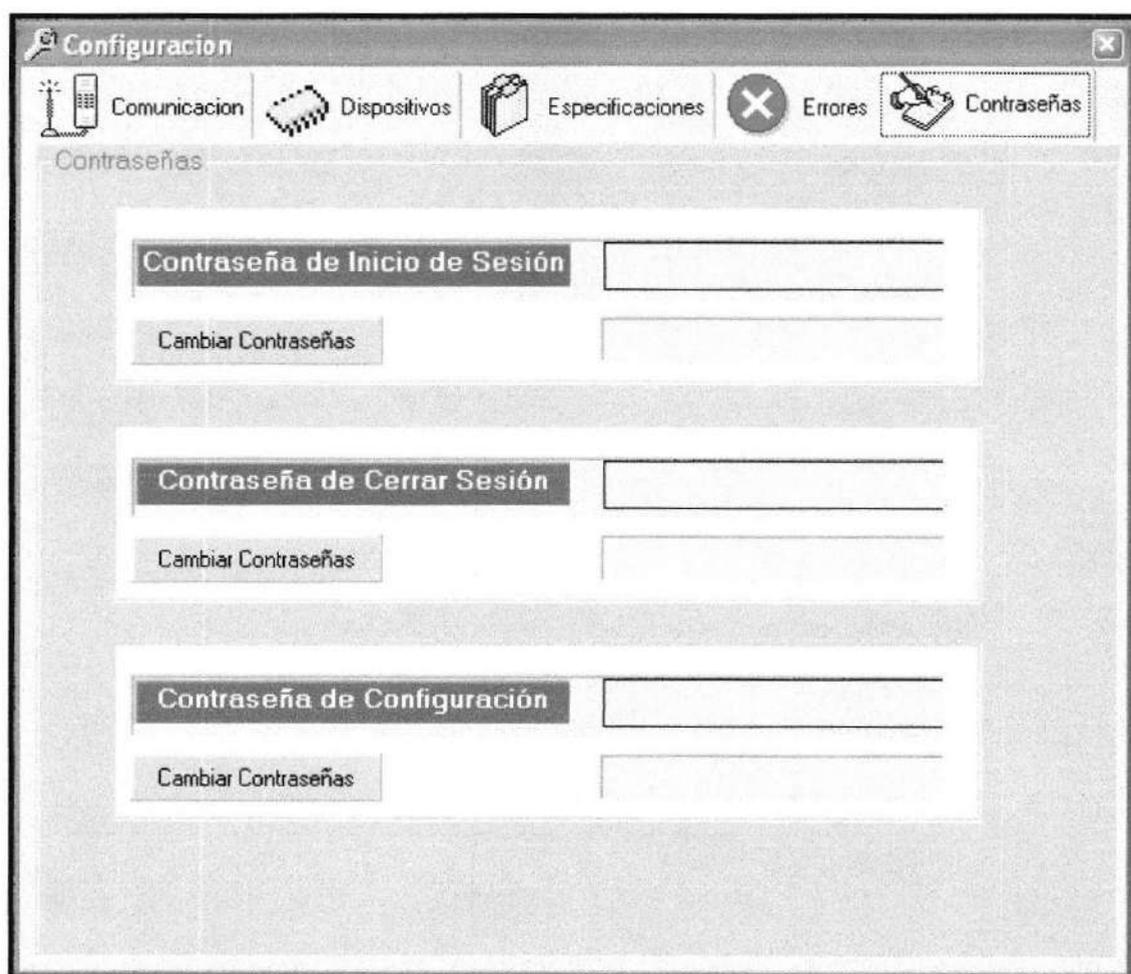


Figura 3.21.- Configuración, etiqueta Contraseñas, las contraseñas son Inicio de Sesión, Cerrar Sesión y Contraseña de Configuración.

### 3.3 CARACTERISTICAS ESPECIALES

El software sistema de monitoreo, adicional a las pantallas anteriormente mencionadas interactúa con otras aplicaciones y con dispositivos (hardware)

para recolectar la información. Para comunicarse con las tarjetas electrónicas usa el puerto serial comm1, por lo que hay una parte en el software diseñado para controlar la comunicación. Cuando se almacena una gran cantidad de datos para después analizarlos es necesaria una base de datos para guardarlos, organizarlos y para consultarlos en cualquier momento por lo que el software interactúa con una base de datos en Acces. Además interactúa con el sistema operativo para crear archivos o bases de datos por año de producción de baterías. Todas estas características se explicarán en las siguientes secciones.

### **3.3.1 PUERTO SERIAL.**

El puerto serial utilizado es el comm1, el tipo de comunicación con las tarjetas de adquisición de datos es uno de los más normalizados, como es vía RS-232, la recolección y tratamiento de datos en el buffer de entrada es realizado en Visual Basic.

### 3.3.2 BASES DE DATOS.

Las bases de datos son muy usadas hoy en día en diferentes programas que necesitan llevar un control o una supervisión de registros en diferentes aplicaciones.

Con una base de datos tengo una recopilación de información sobre los datos de carga de las líneas de baterías, en las tablas de la base se encuentran toda la información organizada en tablas para que el operador pueda hacer una consulta rápida de los datos de carga de cada línea. Las columnas de cada tabla se les llaman Campos y las filas Registros.

#### *Interacción del sistema de monitoreo con bases de datos en Acces*

En Visual Basic existen controles que permiten la conexión con bases de datos, este acceso el sistema de monitoreo lo hace cada cierto tiempo, una porción del código que permite esto se cita como un ejemplo:

```
Data1.RecordSource = "SELECT * FROM Régimen_Carga"
```

La interacción con Acces se hace mediante código SQL, la instrucción "SELECT" indica los campos que se van a consultar, "\*" indica todos los campos de la tabla y "FROM" indica de que tabla se va a consultar. Régimen\_Carga.Data1.Recordsource es el control que se usa para esta conexión y como ya se mencionó el parámetro Tiempo de monitoreo proporciona el tiempo en que el sistema almacena los datos en Acces.

### *Organización de las bases de datos*

#### *Base de datos Reg.Carga*

Esta Base de Datos se creó para guardar datos del sistema de monitoreo, sirve para organizar la información que le llega al sistema de las líneas de baterías como los datos de líneas calibradas, el orden de las líneas, y otra información que usa el fabricante para cargar sus baterías, como el régimen de carga y ciertas configuraciones propios del programa, que necesita ser almacenada.



En la Figura 3.22 podemos observar las tablas que conforman esta base, además cada tabla tiene un identificador (Id) numérico, es importante para relacionar las tablas y para hacer consultas. Las tablas de esta base son:

- a) Calibración.
- b) Configuraciones.
- c) OrdenLineas.
- d) Régimen\_Carga.

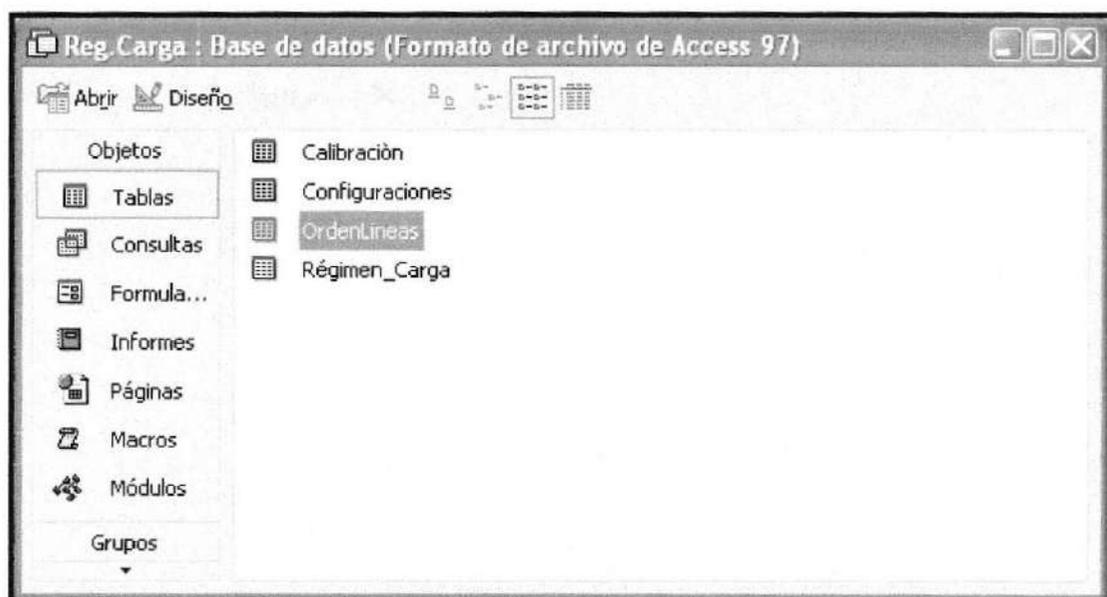


Figura 3.22.- Base de Datos Reg.Carga, sirve para organizar datos del sistema

*a) Calibración*

La tabla Calibración de la Figura 3.23 guarda ciertas calibraciones que se realizaron a las líneas, así podremos consultar en el software estos datos.

Id	mv Maximo en Amperimetro	Corriente Maxima Cargador	Corriente Medida	Tempertaura Medida	Voltaje Medido
5211	0,3	30	0	1	1
5212	0,1	30	0	28,1	0
5213	0,1	30	0	1	1
5214	0,1	30	0	145	45
5215	0,1	30	0	0	0
5216	0,2	30	0	56	56
5217	0,2	30	0	0	0
5218	0	0	0	0	0
5219	0	0	0	0	0
5220	0	0	0	0	0
5221	0	0	0	0	0
5222	0	0	0	0	0
5223	0	0	0	0	0
5224	0	0	0	0	0
5225	0	0	0	0	0
5226	0	0	0	0	0
5227	0	0	0	0	0
5228	0	0	0	0	0
5229	0	0	0	0	0
5230	0	0	0	0	0
5231	0	0	0	0	0
5232	0	0	0	0	0

Registro: de 200

Figura3.23.- Tabla Calibración de la base de datos Reg.Carga

### b) Configuraciones

Aquí se guardan valores de configuración del programa como el tiempo de monitoreo, adquisición de datos, Alta temperatura, etc. y en la otra columna se guardan las contraseñas de seguridad del sistema (Figura 3.24).

Id	Valores del Sistema	Contraseñas
1	0	
2	1	
3	7	
4	0	
5	0	
* (Autonumérico)	0	

Figura3.24.- Tabla Configuraciones, almacena valores del sistema y Contraseñas

### c) OrdenLineas

En esta tabla (Figura 3.25) realiza la consulta la pantalla Configuración, etiqueta Dispositivos, debido a que guarda los datos de código de dispositivo, ubicación y la línea que se la ha asignado.

Id	Dispositivo	Ubicación del Dispositivo	Datos por Dispositivo	Desde Línea
1			0	88
2			0	2
3			0	0
4			0	9
5			0	0
6			0	0
7			0	0
8			0	0
9			0	0
10			0	0
11			0	0
12			0	0
13			0	0
14			0	0
15			0	0
16			0	0
17			0	0
18			0	0

Registro: 1 de 128

Figura 3.25.- Tabla OrdenLineas

#### d) Régimen\_Carga

La tabla régimen de carga de la Figura 3.26 almacena toda la información de las especificaciones de carga de cada tipo de batería, los campos de la tabla son Tipo de Batería, N° Placa+, N° Placa-, Cantidad de amperios horas, amperaje de carga y horas de carga.

Id	Tipo de Batena	Nº Placa +	Nº Placa -	Cantidad de Ah	Amperaje de Carga	Horas de Carga
1	6 160 27	14	13	560	14	40
2	8 195 33	17	16	0.1	17	40
3	12 38 9	5	4	152	5	30
*	(Autonumérico)	0	0	0	0	0

Registro: 1 de 3

Figura 3.26.- Tabla Régimen\_Carga.

#### *Base de Datos Archivo200X*

La base de datos archivo200X almacena los datos de todo el proceso de carga (Figura 3.27), el programa crea una base de datos por año, donde 200X es el año actual, existe una tabla por cada por cada línea de baterías en este caso son 200 líneas de baterías, en cada tabla se guarda los datos de esa línea durante un año.

A medida que una línea de baterías está cargando, el programa cada tiempo de monitoreo guarda los valores de carga en la base de datos. En esta base de

datos existen la tabla a) Estadístico hace un cálculo estadístico y b) Líneas de 1 a 200. A continuación se explica los campos y registros de los dos tipos de tabla.

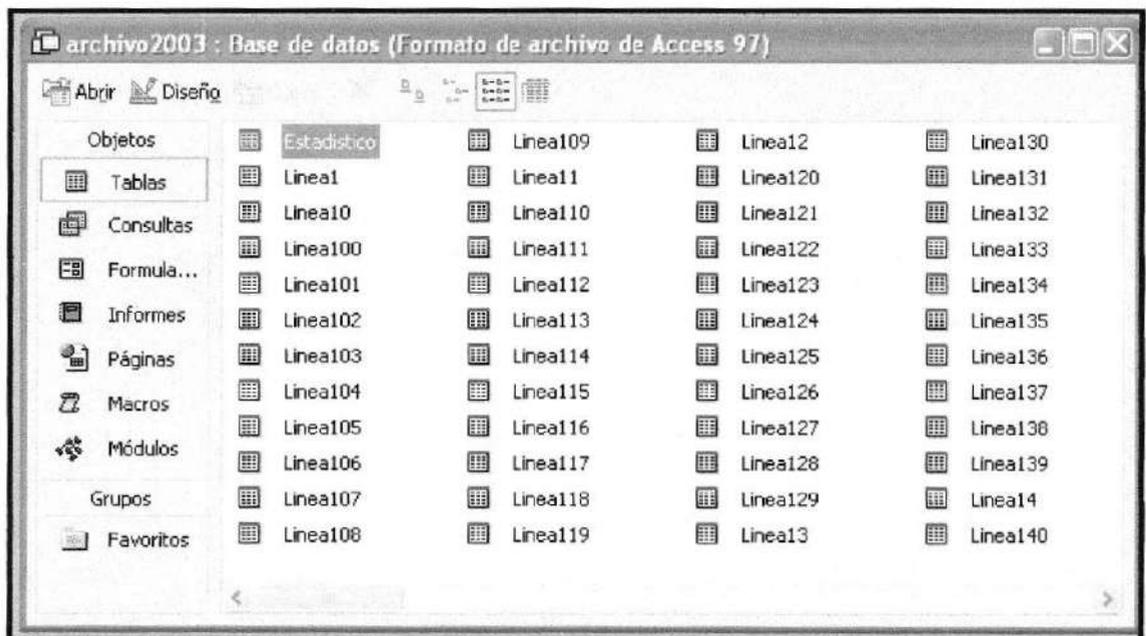


Figura3.27.- Base de datos Archivo2003, esta base se crea Automáticamente al instalar el programa

#### a) Estadístico

En la tabla de la figura 3.28 hay información para que el programa calcule cuantas baterías ha cargado en el mes según el tipo o modelo, en el primer

campo como en todas las tablas está el identificador numérico, en el segundo campo se encuentra el tipo de batería cargada, el tercer campo la fecha de la carga y el cuarto campo el número de baterías de cada tipo de batería cargadas.

Cuando el programa hace la consulta de algún tipo de batería busca en el segundo campo y luego suma el número de baterías cargadas del mismo tipo de batería de la consulta.

Estadístico : Tabla				
	Id	Tipo de Bateria	Fecha	NúmeroBaterias
▶	212	8/195/33	04/01/2003	9
	213	8/195/33	04/02/2003	8
	214	8/195/33	04/03/2003	45
	215	8/195/33	04/04/2003	23
	216	8/195/33	04/05/2003	27
	217	8/195/33	04/06/2003	28
	218	8/195/33	05/07/2003	10
	219	8/195/33	05/08/2003	15
	220	8/195/33	05/09/2003	25
	221	8/195/33	05/10/2003	19
	222	8/195/33	05/10/2003	45
	223	8/195/33	05/10/2003	5
*	(Autonumérico)			0

Registro: 1 de 12

Figura3.28.- Tabla Estadístico

### b) LíneaX



La tabla LíneaX de la figura 3.29 almacena información del proceso de carga de la línea de baterías donde X es el número de la línea, esta tabla es usada para las consultas que realizan las pantallas de gráficos y tabla de datos, ya que aquí se encuentran, todas las variables monitoreadas por el sistema: corriente, temperatura, voltaje, energía y capacidad, tipos de baterías con el estado de

carga: Cargando, Línea Cargada y Sin Cargar, fecha y horas de carga con el código de archivo correspondiente. El parámetro IdLínea es importante, está en forma ascendente y se lo usa para graficar las variables.

IdLínea	TipoBateria	Estado	Fecha	Horas de Carga	Voltaje (V)	Corriente (A)	Temperatura (°C)	Energía (KWh)	Capacidad (Ah)	Archivo
7	8/195/33	Cargando	04/09/2003	0 03:02	0	4,2	32,7	0	0,21	04/09/2003 17:00
8	8/195/33	Cargando	04/09/2003	0 04:01	0	4,2	32,7	0	0,28	04/09/2003 17:00
9	8/195/33	Cargando	04/09/2003	0 05:00	0	4,2	32,7	0	0,35	04/09/2003 17:00
10	8/195/33	Cargando	04/09/2003	0 05:59	0	4,2	32,7	0	0,42	04/09/2003 17:00
11	8/195/33	Cargando	04/09/2003	0 06:59	0	4,2	32,7	0	0,49	04/09/2003 17:00
12	8/195/33	Cargando	04/09/2003	0 07:58	0	4,2	32,7	0	0,56	04/09/2003 17:00
13	8/195/33	Cargando	04/09/2003	0 08:58	0	4,2	32,7	0	0,62	04/09/2003 17:00
14	8/195/33	Cargando	04/09/2003	0 09:57	0	4,2	32,7	0	0,69	04/09/2003 17:00
15	8/195/33	Cargando	04/09/2003	0 10:56	0	4,2	32,7	0	0,76	04/09/2003 17:00
16	8/195/33	Cargando	04/09/2003	0 14:54	0	4,2	32,7	0	1,04	04/09/2003 17:00
17	8/195/33	Cargando	04/09/2003	0 15:38	0	3,45	32,7	0	1,09	04/09/2003 17:00
18	8/195/33	Cargando	04/09/2003	0 15:42	0	4,12	32,7	0	1,09	04/09/2003 17:00
19	8/195/33	Cargando	04/09/2003	0 16:02	0	3,74	32,7	0	1,12	04/09/2003 17:00
20	8/195/33	Cargando	04/09/2003	0 16:05	0	3,74	32,7	0	1,12	04/09/2003 17:00
21	8/195/33	Cargando	04/09/2003	0 16:22	0	1,84	32,7	0	1,14	04/09/2003 17:00
22	8/195/33	Cargando	04/09/2003	0 16:26	0	3,93	32,7	0	1,14	04/09/2003 17:00
23	8/195/33	Cargando	04/09/2003	0 17:05	0	4,11	32,2	0	1,18	04/09/2003 17:00
24	8/195/33	Cargando	04/09/2003	0 18:04	0	4,12	32,2	0	1,25	04/09/2003 17:00
25	8/195/33	Cargando	04/09/2003	0 19:04	0	4,19	32,7	0	1,32	04/09/2003 17:00
26	8/195/33	Cargando	04/09/2003	0 20:03	0	4,19	32,7	0	1,39	04/09/2003 17:00
27	8/195/33	Cargando	04/09/2003	0 21:02	0	4,19	32,7	0	1,46	04/09/2003 17:00

Figura 3.29.- Tabla Línea1

### 3.3.3 ARCHIVOS ANUALES.

El sistema de monitoreo usa dos bases de datos, uno para almacenar datos del mismo sistema y el otro para datos de Carga, al momento de instalación del software crea la carpeta Sistema de Monitoreo, y con ella la base de dato modelo Sistema\_MonitoreoAcces, ya que el programa crea las bases de datos

para cada año tomando como modelo la ya mencionada base modelo, como se puede apreciar en la Figura 3.30.

Al iniciar por primera vez el programa en una PC el sistema crea una base de datos con el nombre archivoXXXX, donde las equis son el año actual en la que se almacenarán los datos de carga, cuando se inicia un nuevo año el sistema ahora los guardará en la base de datos del nuevo año que inició.



Figura 3.30.- Carpeta Sistema de Monitoreo con las bases de datos.

# **CAPITULO 4**

## **4.- DESCRIPCION DE LOS ELEMENTOS DEL SISTEMA.**

### **4.1 MICROCONTROLADORES PIC 16F877A.-**

Los microcontroladores están conquistando el mundo. Están presentes en nuestro trabajo, en nuestra casa y en nuestra vida en general. Se pueden encontrar controlando el funcionamiento de ratones y teclados en los computadores, en los teléfonos, en los hornos microondas y en los televisores de nuestro hogar. La invasión de microcontroladores comenzó hace muy poco y en el siglo XXI se consolidará. La casi totalidad de los automatismos y

electrodomésticos que hoy conocemos, serán gobernados por microcontroladores. Las extensas áreas de aplicación de los microcontroladores, que pueden considerarse ilimitadas, exigirán un gigantesco trabajo de diseño y fabricación, por parte del Ingeniero Electrónico.

#### **4.1.1 EL MICROCONTROLADOR**

Es un circuito integrado programable que contienen todos los componentes de un computador, se emplea para controlar el funcionamiento de una tarea determinada y debido a su reducido tamaño, suele ir incorporado en el propio dispositivo al que gobierna (Figura 4.1).

En su memoria sólo reside un programa destinado a gobernar una aplicación determinada, sus líneas de entrada/salida soportan el conexionado de los sensores y actuadores del dispositivo a controlar y todos los recursos complementarios disponibles tienen como única finalidad atender sus requerimientos. Una vez programado y configurado el microcontrolador solamente sirve para gobernar la tarea asignada.

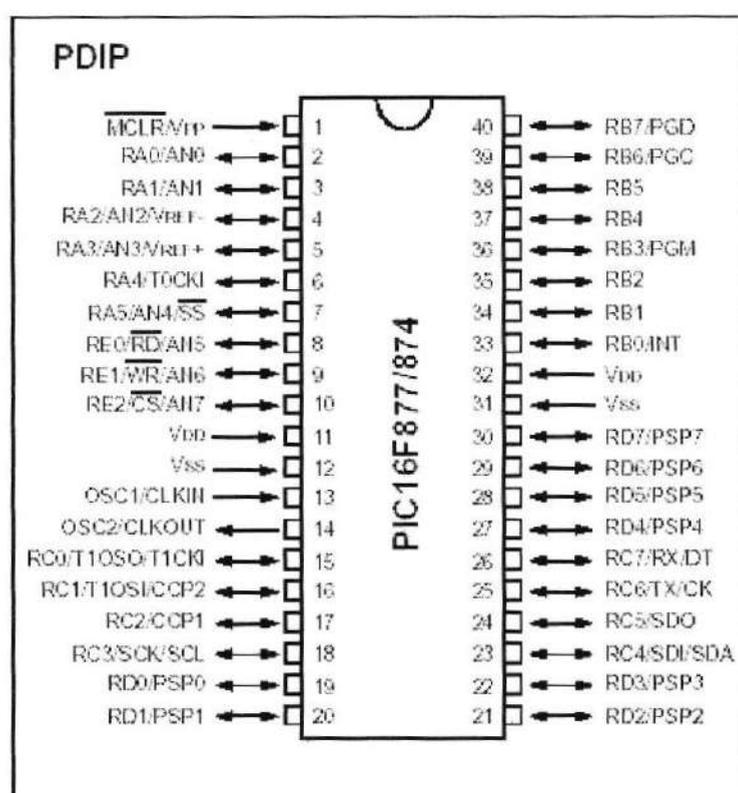


Figura 4.1.- Pic16F877A, es la serie de los Pics en este proyecto.

#### 4.1.2 PIC16F877A Y SUS PRESTACIONES.

La subfamilia Pic16F87x es un conjunto de cuatro modelos de Pic que tienen una memoria FLASH que refuerzan a la gama de Pics 16F84, son los modelos

Pic 16F873/4/6/7A, que se encuadran con la gama media 16Cxxx de microcontroladores de 8 bits, en la tabla I se puede observar una comparación entre los cuatro modelos y el PIC16F84.

Tabla I

Cuadro Comparativo cuatro tipos de Pic's gama 16F87x

MODELO	PIC16F84A	PIC16F873	PIC16F874	PIC16F876	PIC16F877A
Mem. Bytes Prog. (Flash )	1792	7168	7168	14336	14336
Pala_Bras	1024x14	4096x14	4096x14	8192x14	8192x14
Mem. Bytes EEP ROM	64	128	128	256	256
Datos Bytes RAM	68	192	192	368	368
C A/D BOD	No	5 (10 bits)	8 (10 bits)	5 (10 bits)	8 (10 bits)
(Detección de baja tensión)	No	Sí	Sí	Sí	Sí
Línea E/S Comunicación Serie	13	22	33	22	33
CCP	No	USART/ MSSP	USART/ MSSP	USART/ MSSP	USART/ MSSP
Temporiza_Dores	No	2	2	2	2
	1-8 bit, 1-WDT	1-16 bit, 2-8 bit, 1-WDT			
FREC.Max. MHz	20	20	20	20	20
ICSP (Programación Serie en Circuito)	Sí	Sí	Sí	Sí	Sí

Encapsulados	18P, 18SO, 20SS	28SP, 28SO	40P, 44L, 44PQ, 44PT	28SP, 28SO	40P, 44L, 44PQ, 44PT
Fuentes de Interrupción Comunicación Paralelo	4	13	14	13	14
	No	No	Sí	No	Sí

Los nuevos PIC de la familia 16F877A estuvieron en el mercado alrededor del año 2000 y tiene un sin numero de aplicaciones a continuación se muestra la tabla II con alguna de sus aplicaciones.

Tabla II  
Aplicaciones con los Pic16F877A.

<b>Aplicaciones</b>			
<b>Consumo</b>	<b>Automoción</b>	<b>Telecomunicaciones</b>	<b>Industria</b>
Equipo de Tv	Sistema de seguridad del vehículo	Teléfono celular	Control de Motores
Receptor de Radio	Detector de radar	Teléfono inalámbrico	Compresor
Reproductor de CD	ABS	identificador de Llamadas	Termostato
Control Remoto	Velocímetro	Módem	Utilidades
video Consola	Climatizador		De medida
Cámara	Inyección de Combustible		Robótica
Mando de Garaje	Suspensión activa		Procesos De Control
Detector de Monóxido de Carbono	Sensor de airbag		Detector de humo
Microondas			Lector de Tarjetas
Lavadora			
Secadora			
Aparatos de Cocina			
Aspiradoras			

El conjunto de Microcontroladores consta de 4 gamas:

- 1) Gama Básica: Pic12c5xx y Pic16c5xx con un repertorio de 33 instrucciones de 12 bits y dos niveles de pilas.

- 2) Gama Media: familias PIC 12c6xx y Pic16cxx con un repertorio de 35 instrucciones de 14 bits, ocho niveles de Pila y un vector de interrupción.
- 3) Gama Alta: familia Pic17cxxx, con un repertorio de 58 instrucciones de 16 bits, con 16 niveles de Pila y 4 vectores de interrupción.
- 4) Gama Mejorada: familia Pic18cxxx con 77 instrucciones de 16 bits ,32 niveles de Pila y 4 vectores interrupción.

#### *Características del Pic16F877A*

El PIC16F877A posee una memoria de tipo Flash que le permite borrar y grabar programas eléctricamente y tiene una capacidad 368 bytes en la RAM, además tienen 4 bancos de 128 bytes cada uno se selecciona con los bits RP0 y RP1 del Registro Estado.

El siguiente cuadro (Tabla III) muestra sus características:

Tabla III

Características del Pic16F877A

<b>CARACTERÍSTICAS</b>	<b>PIC16F877A</b>
Frecuencia de Operación	DC - 20 MHz
Memoria de Programa tipo FLASH	4K
Memoria de Datos RAM	368
Causas de RESET	POR, BOR (PWRT, OST)
Memoria de datos EEPROM	256
Interrupciones	14
Puertas E/S	Puertas A,B,C,D,E
Temporizadores	3
Modulo Captura/Comparación y PWM	2
Comunicación Serial	MSSP, USART
Comunicación Paralelo	PSP
Modulo Analógico-Digital	8 canales, Resolución 10 bits
Juego de Instrucciones	35 instrucciones

El PIC16F877A tiene 5 Puertos bidireccionales y algunos pines de estas puertas son multiplexadas con otras funciones, las cuales se configuran en algunos registros internos, los cuales manejan las características especiales como son:

- 1) Modulo CCP.- Capturan y comparan pulsos aplicados en uno de los pines de la Puerta B. La captura se efectúa con una precisión de 1,25 ns y una resolución de 16 bits, mientras que la comparación con igual resolución alcanza una precisión de 200 ns además tiene una modulación de Ancho de Pulso PWM usado para el control de Motores
  
- 2) Comunicación Serie.- El PIC16F877A soporta la comunicación serie USART la cual es utilizada más para comunicación entre subsistemas o máquinas como un ordenador con un protocolo muy utilizado como es el RS-232 , la otra comunicación serie es MSSP, destinada a la comunicación entre diversos circuitos integrados y que admite los protocolos I2C y SPI.



CIB-ESPOL

- 3) Comunicación Paralelo.-Tiene disponible el protocolo PSP, el cual es más rápido que la comunicación serie, pero la desventaja es que ocupa muchas líneas de E/S, 8 de la Puerta D y 3 para el control de la Puerta E.
  
- 4) Conversor Analógico-Digital.- Posee una resolución de 10 bits con 8 canales de adquisición de datos.

#### **4.1.3 ARQUITECTURA INTERNA**

Los microcontroladores de la familia 16F87xx tiene un número de características comúnmente encontrado en los microprocesadores RISC a continuación se enumeran algunas características:

- Arquitectura Harvard
- Simple juego de instrucciones
- Un solo ciclo de Instrucciones
- Juego reducido de Instrucciones
- Arquitectura de Registros
- Instrucciones simétricas

#### *Arquitectura Harvard*

La característica de la Arquitectura Harvard (Figura 4.3) es que tiene la memoria de código y la memoria de datos separadas y se accede a ellas por buses de datos distintos esto mejora el trabajo con las dos memorias, aun mejor que la tradicional arquitectura de Von Neumann en la cual el código y los datos se encuentran en la misma memoria por lo que para ejecutar una instrucción tiene que acceder a la memoria a través del mismo bus más de una vez por lo que el bus estaría muy congestionado. Mientras que con la arquitectura Harvard, la instrucción es ejecutada en un solo ciclo de instrucción.

Mientras la memoria de código esta siendo accedida, la memoria de datos está en un bus independiente y puede ser leída y escrita. Estos buses separados permiten que una instrucción se ejecute mientras que la siguiente instrucción es buscada.

Una comparación entre Arquitectura Harvard y Von Neumann se puede ver en la siguiente Figura 4.2:

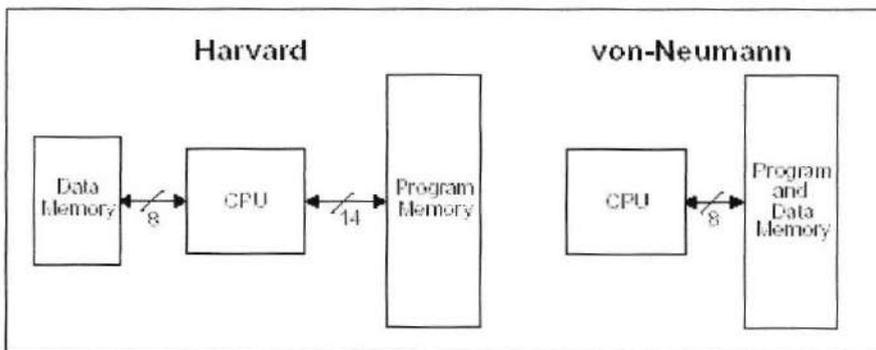


Figura 4.2.- Diferencias entre Arquitectura Harvard y Von-Neuman  
Instrucciones de mayor longitud

La longitud de las instrucciones tiene más bits por instrucción más que los bus de datos de 8 bits, esto es posible debido a la separación de los dos buses, que permite un uso más eficiente de la memoria de código.

#### *Un solo ciclo de instrucciones*

Siendo la memoria de código de 14 bits, la instrucción completa es buscada en un solo ciclo (TCY). La instrucción contiene toda la información requerida y es ejecutada en un ciclo. Podría haber un retardo en la ejecución si el resultado de la instrucción ha cambiado el contenido del Contador del Programa (PC).

#### *Juego reducido de instrucciones*

Cuando un juego de instrucciones es bien diseñado y altamente simétrico son requeridas pocas instrucciones para ejecutar todas las tareas necesitadas, esto es lo que sucede con el Pic16f877a y además permite que todas las instrucciones puedan ser aprendidas más rápidamente.

### *Arquitectura de Registros*

Los registros y la memoria de datos pueden ser direccionadas directa o indirectamente. En los registros se pueden configurar los bits para ejecutar funciones especiales.

### *Instrucciones Simétricas*

Instrucciones simétricas hacen posible llevar a cabo alguna operación en algún registro usando cualquier método de direccionamiento.



#### **4.1.4 ORGANIZACIÓN DE LA MEMORIA Y REGISTROS DE CONTROL.**

La memoria FLASH en la que se graba el programa de los Pic16f87x, puede tener una capacidad de 4k u 8k palabras de 14 bits cada una. La memoria está dividida en páginas de 2k palabras y está direccionada con el PC que tiene un tamaño de 13 bits como se ve en la Figura 4.4. Hay una Pila que tiene 8 niveles de profundidad, funciona automáticamente y no dispone de instrucciones para guardar o sacar de ella información.

Existen algunas instrucciones como CALL y las interrupciones el valor del PC se carga en el nivel superior de la pila y con RETURN, RETFIE y RETLW el valor contenido en el nivel superior de la pila se carga en el PC. Hay que tener cuidado ya que la pila tienen 8 niveles y no hay que sobrepasarlo con anidamientos en las subrutinas. El vector RESET ocupa la dirección 0000h y el vector de interrupción 0004h.

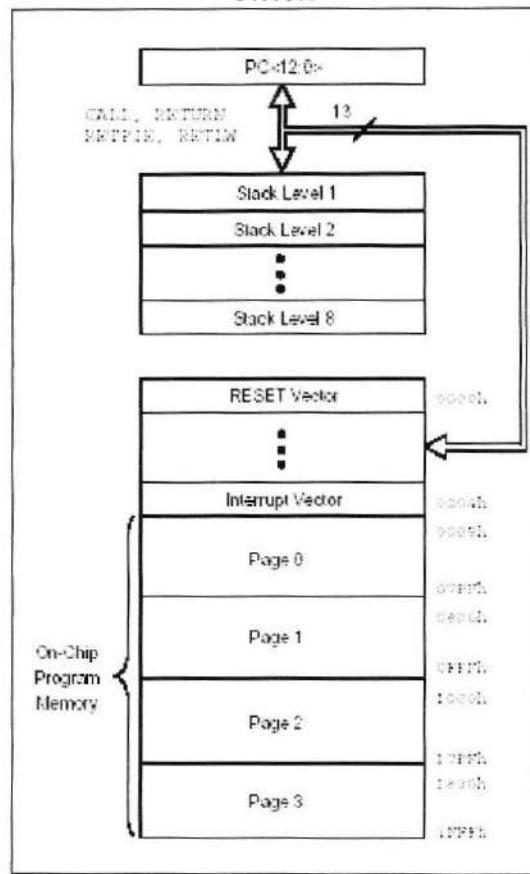


Figura 4.4.- Organización de la memoria Tipo Flash

### *Memoria de datos RAM*

La memoria de datos esta posicionada en múltiples posiciones en la RAM y en la EEPROM. En la sección de la RAM se encuentran registros que gobiernan el funcionamiento del procesador y el manejo de todos sus periféricos, además existen Registros de Propósito General donde se guarda algún tipo de información propia de la aplicación y la EEPROM donde se guarda datos de forma no volátil, es decir que no se borran así se produce un reset en el dispositivo.

La memoria de datos se encuentra en múltiples bancos de memoria en el PIC 16F877A son 4 bancos de 128 bytes cada uno. Los bits RP1 y RP0 del registro STATUS se usan para acceder a la memoria de Datos, la tabla IV muestra los valores de RP1 y RP0.

Tabla IV.

Valores que toman los bits RP1:RP0 para acceder a los 4 bancos de memoria

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

En la figura 4.5 se muestra los 4 bancos de la RAM con los nombres de los registros.

File Address	File Address	File Address	File Address
Indirect addr. <sup>(1)</sup> 00h	Indirect addr. <sup>(1)</sup> 80h	Indirect addr. <sup>(1)</sup> 100h	Indirect addr. <sup>(1)</sup> 180h
TMR0 01h	OPTION REG 81h	TMR0 101h	OPTION REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h		
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PORTC 07h	TRISC 87h		
PORTD <sup>(1)</sup> 08h	TRISD <sup>(1)</sup> 88h		
PORTE <sup>(1)</sup> 09h	TRISE <sup>(1)</sup> 89h		
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah
INTCON 0Bh	INTCON 8Bh	INTCON 10Bh	INTCON 18Bh
PIR1 0Ch	PIE1 8Ch	EEDATA 10Ch	ECON1 18Ch
PIR2 0Dh	PIE2 8Dh	EEADR 10Dh	ECON2 18Dh
TMR1L 0Eh	PCON 8Eh	EEDATH 10Eh	Reserved <sup>2</sup> 18Eh
TMR1H 0Fh		EEADRH 10Fh	Reserved <sup>2</sup> 18Fh
T1CON 10h			
TMR2 11h	SSPCON2 91h		
T2CON 12h	PR2 92h		
SSFBUF 13h	SSPADD 93h		
SSPCON 14h	SSPSTAT 94h		
CCPR1L 15h			
CCPR1H 16h			
CCPR1CON 17h			
RCSTA 18h	TXSTA 95h	General Purpose Register 16 Bytes	General Purpose Register 16 Bytes
TXREG 19h	SPBRG 96h		
RCREG 1Ah			
CCPR2L 1Bh			
CCPR2H 1Ch			
CCPR2CON 1Dh			
ADRESH 1Eh	ADRESL 9Eh		
ADCON0 1Fh	ADCON1 9Fh		
General Purpose Register 9C Bytes	General Purpose Register 80 Bytes	General Purpose Register 80 Bytes	General Purpose Register 80 Bytes
	addresses 70h-7Fh	addresses 70h-7Fh	addresses 70h-7Fh
Bank 0 7Fh	Bank 1 FFh	Bank 2 17Fh	Bank 3 1FFh

Unimplemented data memory locations, read as 0.  
<sup>1</sup> Not a physical register.

**Note** 1: These registers are not implemented on the PIC16F870.  
 2: These registers are reserved, maintain these registers clear.

Figura 4.5.- Distribución de la memoria RAM.



Para utilizar adecuadamente los periféricos y funciones especiales del PIC hay que conocer la misión de cada uno de los Registros.

### *Registros de Control de Memoria*

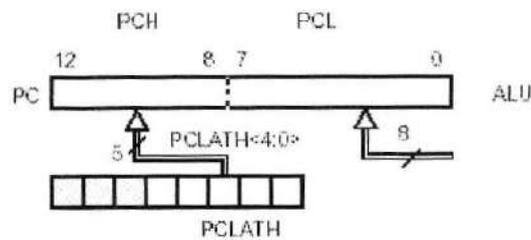
#### *PCL y PCLATH*

Los registros PCL y PCLATH son registros para el control de la memoria del programa. En ellos se encuentran los bits del PC<13:0> (Contador del Programa). En PCL se encuentran los 8 bits menos significativos PC<8:0> y pueden ser leídos y escritos. Los bits más significativos de PC<12:8> no pueden ser accedidos directamente pero sí indirectamente a través de PCLATH.

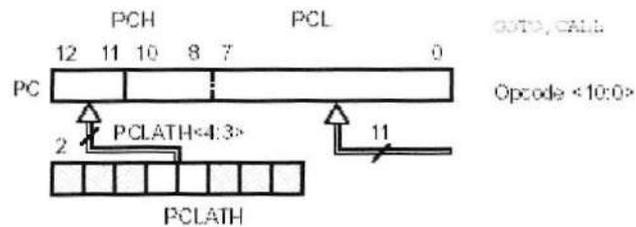
Con las instrucciones de salto CALL y GOTO proporcionan 11 bits de la dirección a saltar por lo que el salto está limitado a 2k por lo que si se cambia de

banco hay que programar correctamente los bits PCLATH<4:3> para cambiar de banco en dichas instrucciones.

Se carga el PC.



Se carga el PC con las Instrucciones CALL y GOTO.



#### Direccionamiento directo e indirecto

Para direccionar la memoria RAM se usa los registros STATUS para el direccionamiento directo y FSR para el direccionamiento indirecto con los 7 bits de menos peso señalan la dirección y el bit de más peso señalan el banco conjuntamente con el bit IRP del Registro de Estados (STATUS<7>).

El registro INDF no está físicamente implementado pero se puede escribir o leer con el direccionamiento indirecto, por lo tanto se pueden guardar datos en la RAM mediante este método y para los registros de funciones especiales se puede usar un direccionamiento directo a través del registro STATUS. La Figura 4.6 muestra ambos direccionamientos:

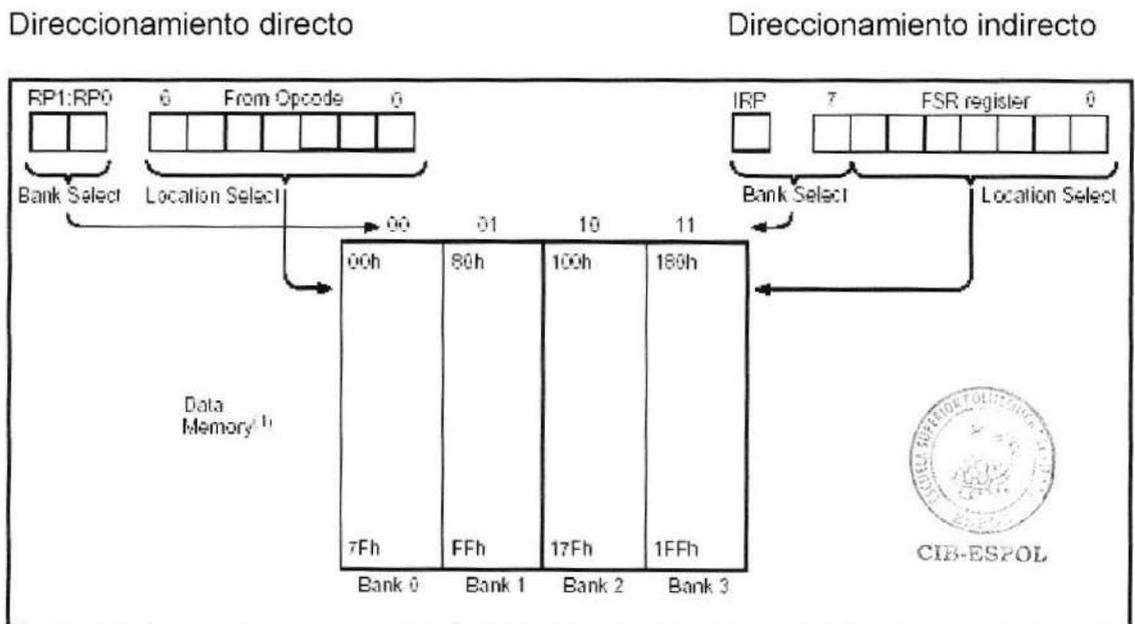


Figura 4.6.- Direccionamiento directo e indirecto para acceder a la memoria de datos.

El siguiente ejemplo limpia las posiciones de 20 a 2F usando un direccionamiento Indirecto.

```

MOVW 0x20 ;initialize pointer
MOVWF POR ;to RAM
NEXT CLRF INDF ;clear INDF register
      INCF POR,F ;inc pointer
      BTSC POR,4 ;all done?
      GOTO NEXT ;no clear next
CONTINUE
      : ;yes continue

```

El siguiente ejemplo muestra el cambio de banco en un direccionamiento directo.

```

CLRF STATUS ; Clear STATUS register (Bank0)
:
:
BCF STATUS, RP0 ; Bank1
:
:
BCF STATUS, RP0 ; Bank0
:
:
MOVW 0x80 ; Set RP0 and RP1 in STATUS register; other
XORWF STATUS, F ; Bits unchanged (Bank?)
:
:
BCF STATUS, RP0 ; Bank2
:
:
BCF STATUS, RP1 ; Bank0

```

### *Principales Registros de Control*

La unidad central de procesamiento (CPU) es responsable del uso de la información en la memoria de código para controlar la operación del dispositivo.

Muchas de estas instrucciones operan en la memoria de datos. Para operar en dicha memoria, es requerida la ALU (Unidad Lógica Aritmética). Además de ejecutar operaciones aritméticas y lógicas la ALU controla los bits de estado en el Registro de Estado (STATUS). El resultado de algunas instrucciones cambia el estado de dichos bits.

En síntesis el CPU reconoce el código de operación de las instrucciones. La tabla es un cuadro resumido de instrucciones:

Tabla V  
Instrucciones del Pic16F877A

Mnemonic, Operands	Description	Cycles	14-Bit Instruction Word				Status Bits Affected	Notes	
			MSb	LSb					
<b>BYTE-ORIENTED FILE REGISTER OPERATIONS</b>									
ADDWF	f,d	Add W and f	1	00	0111	0111	1111	C,DC,Z	1,2
ANDWF	f,d	AND W with f	1	00	0101	0111	1111	Z	1,2
CLRF	f	Clear f	1	00	0001	1111	1111	Z	2
CLRW	-	Clear W	1	00	0001	0000	0000	Z	
COMF	f,d	Complement f	1	00	1001	0111	1111	Z	1,2
DECF	f,d	Decrement f	1	00	0011	0111	1111	Z	1,2
DECFSZ	f,d	Decrement f, Skip if 0	1(2)	00	1011	0111	1111		1,2,3
INCF	f,d	Increment f	1	00	1010	0111	1111	Z	1,2
INCFSZ	f,d	Increment f, Skip if 0	1(2)	00	1111	0111	1111		1,2,3
IORWF	f,d	Inclusive OR W with f	1	00	0100	0111	1111	Z	1,2
MOVF	f,d	Move f	1	00	1000	0111	1111	Z	1,2
MOVWF	f	Move W to f	1	00	0000	1111	1111		
NOP	-	No operation	1	00	0000	0000	0000		
RLF	f,d	Rotate Left f through Carry	1	00	1101	0111	1111	C	1,2
RRF	f,d	Rotate Right f through Carry	1	00	1100	0111	1111	C	1,2
SUBWF	f,d	Subtract W from f	1	00	0010	0111	1111	C,DC,Z	1,2
SWAPF	f,d	Swap nibbles in f	1	00	1110	0111	1111		1,2
XORWF	f,d	Exclusive OR W with f	1	00	0110	0111	1111	Z	1,2
<b>BIT-ORIENTED FILE REGISTER OPERATIONS</b>									
BCF	f,b	Bit Clear f	1	01	000b	1111	1111		1,2
BSF	f,b	Bit Set f	1	01	010b	1111	1111		1,2
BTFSC	f,b	Bit Test f, Skip if Clear	1(2)	01	100b	1111	1111		3
BTFSS	f,b	Bit Test f, Skip if Set	1(2)	01	110b	1111	1111		3
<b>LITERAL AND CONTROL OPERATIONS</b>									
ADDLW	k	Add literal and W	1	11	111k	000k	000k	C,DC,Z	
ANDLW	k	AND literal with W	1	11	1001	000k	000k	Z	
CALL	k	Call subroutine	2	10	000k	000k	000k		
CLRWDI	-	Clear Watchdog Timer	1	00	0000	0110	0100	TG,PD	
GOTO	k	Go to address	2	10	000k	000k	000k		
IORLW	k	Inclusive OR literal with W	1	11	1000	000k	000k	Z	
MOVLW	k	Move literal to W	1	11	000k	000k	000k		
RETFIE	-	Return from interrupt	2	00	0000	0000	1001		
RETLW	k	Return with literal in W	2	11	0100	000k	000k		
RETD	-	Return from Subroutine	2	00	0000	0000	1000		
SLEEP	-	Go into standby mode	1	00	0000	0110	0011	TG,PD	
SUBLW	k	Subtract W from literal	1	11	110k	000k	000k	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	000k	000k	Z	

### *Unidad Central de Procesamiento*

El CPU se lo puede imaginar como el cerebro del dispositivo, es el encargado de buscar la instrucción a ejecutar, decodificarla y luego ejecutarla.

El CPU muy a menudo trabaja conjuntamente con la ALU para completar la ejecución de la instrucción, además el CPU controla el direccionamiento de la memoria de código y el direccionamiento de la memoria de datos y el acceso a la pila.

### *Ciclos del reloj*

Cada ciclo de instrucción ( $T_{cy}$ ) es acompañado de 4 ciclos Q (Q1-Q4). El tiempo de cada ciclo es el mismo tiempo que tiene el oscilador del dispositivo ( $T_{osc}$ ).

Los ciclos Q proporcionan el tiempo para la decodificación, lectura, procesamiento del dato, escritura, etc. Los 4 ciclos Q que conforman un ciclo de instrucción ( $T_{cy}$ ) puede ser generalizado como:

Q1: Ciclo de decodificación de la instrucción

Q2: Ciclo de lectura del Dato

Q3: Procesamiento del dato

Q4: Ciclo de escritura del Dato.

En la figura 4.7 se observa cada ciclo:

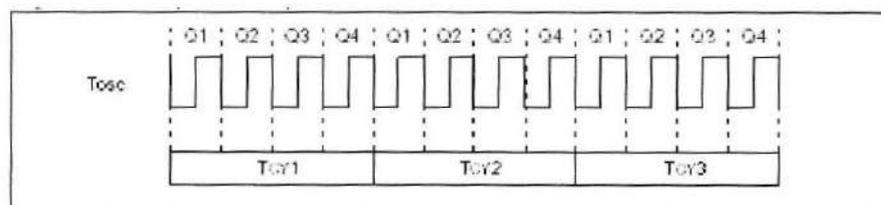


Figura 4.7.- Se muestra un ciclo  $T_{cy}$ , que consta de 4 ciclos Q.

### *Unidad Lógica Aritmética (ALU)*

La ALU es una unidad lógica y aritmética de propósito general. Ejecuta funciones aritméticas y Booleanas entre el dato en el registro de trabajo (W) y algún otro registro.

La ALU tiene un tamaño de 8 bits y es apto para operaciones lógicas, adición, substracción (Figura 4.8). Utiliza dos operandos, uno es el registro de trabajo W y el otro es un registro o una constante. Dependiendo de la instrucción que se ejecuta la ALU podría afectar el estado de los bits Acarreo (C), acarreo en el 4° bit (DC), y señalizador de Zero (Z) que se encuentran en el registro STATUS.

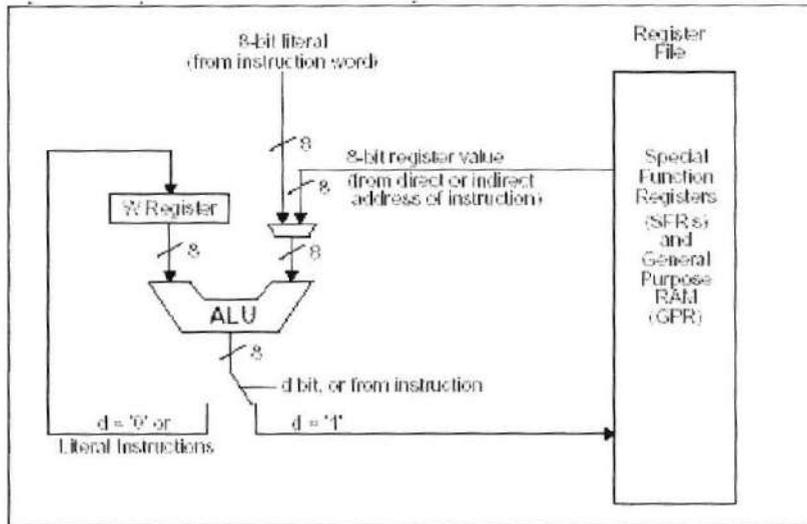


Figura 4.8. - Unidad l3gica Aritm3tica.

### *Registro de Estado (STATUS)*

Sus bits est1n destinados a controlar las funciones vitales del procesador, debido a su importancia est1 duplicado en la cuarta posici3n de los 4 bancos. En el se encuentran el estatus de la ALU, las causas de alg3n RESET, y los bits de selecci3n de los bancos en la memoria de datos.

## REGISTRO STATUS

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	
bit 7								bit 0

Los tres bits de menos peso sirven para determinar el estado de operaciones lógico-aritméticas. Para indicar las causas de un RESET, los bits PD y TO son muy útiles para determinarlo. PD y TO toman el valor de 1 cuando se producen dos eventos especiales *POR* (Power on Reset) que significa que se produce un reset al conectar a la alimentación y *BOR* (Brown on Reset) que significa que se produce un reset cuando el voltaje de la alimentación. Los 3 bits de más peso sirven para seleccionar el banco de la RAM ya sea de forma directa o indirecta.



En ciertas instrucciones el registro estado puede actuar como destino, en cuyo caso los bits Z, C, DC no son escribibles en aquellas instrucciones que les afecte la operación que realizan, por lo tanto instrucciones como `clrf STATUS` solo encera los 3 bits más significativos mientras que los otros quedan sin

cambiar por ese motivo se recomienda utilizar *bcf*, *bcf*, *swapf* y *movf* para manejar los bits de este registro, debido a que no afectan a Z, DC, C.

Para identificar si se realizó un POR o un BOR existen unos señalizadores en el registro PCON en el banco 1 con lo que se puede determinar conjuntamente con TO y PD las causas de un Reset. La tabla 4.6 y la Tabla 4.7 indican la función de cada bit del Registro STATUS y OPTION\_REG.

Tabla VI

## Función de los bits del registro STATUS

<b>C:</b>	Se pone uno cuando hay acarreo en el bit de más peso en instrucciones de suma y se pone a cero cuando hay acarreo en instrucciones de resta.
<b>DC:</b>	Funciona igual que en el acarreo C pero este es en el 4° bit es muy útil para operaciones en BCD.
<b>Z:</b>	Se pone a 1 cuando el resultado es cero.
<b>PD:</b>	Se activa a 0 al ejecutarse una instrucción de SLEEP (modo reposo) y se activa a uno cuando se conecta a la alimentación o cuando se usa la instrucción CLRWDWT (Refresco del perro guardián).
<b>TO:</b>	Se activa a cero al desbordarse el perro guardián y se activa a uno cuando se conecta a la alimentación, al ejecutarse CLRWDWT o SLEEP.
<b>RP0:RP1</b>	Selección del banco en la RAM.
<b>IRP</b>	Conjuntamente con el bit de más peso del registro FSR para el direccionamiento indirecto al elegir el banco.

## REGISTRO OPTION\_REG

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
RBPUP	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
bit 7							bit 0

Tabla VII

Función de los bits del registro OPTION\_REG

<b>PS2:PS1</b>	<p>Valor que indica el valor del Divisor de frecuencia usado para el Temporizador o para el perro guardián</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>TMR0</th> <th>WDT</th> </tr> </thead> <tbody> <tr><td>000</td><td>1:2</td><td>1:1</td></tr> <tr><td>001</td><td>1:4</td><td>1:2</td></tr> <tr><td>010</td><td>1:8</td><td>1:4</td></tr> <tr><td>011</td><td>1:16</td><td>1:8</td></tr> <tr><td>100</td><td>1:32</td><td>1:16</td></tr> <tr><td>101</td><td>1:64</td><td>1:32</td></tr> <tr><td>110</td><td>1:128</td><td>1:64</td></tr> <tr><td>111</td><td>1:256</td><td>1:128</td></tr> </tbody> </table>	Bit	TMR0	WDT	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
Bit	TMR0	WDT																										
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																										
<b>PSA:</b>	<p><b>1</b>= El divisor de frecuencia se asigna al WDT. (Perro guardián).  <b>0</b>= El divisor de frecuencia se asigna al TMR0 (Temporizador).</p>																											
<b>TOSE:</b>	<p>Tipo de flanco en la patita TOCK1.  <b>1</b>= Incremento del TMR0 con flanco descendente.  <b>0</b>= Incremento del TMR0 con flanco ascendente.</p>																											
<b>TOCS:</b>	<p>Tipo de reloj para TMR0  <b>1</b>= Pulsos a través de TOCK1 (contador).  <b>0</b>= Pulsos de reloj interno Fosc/4 (temporizador).</p>																											
<b>INTEDG:</b>	<p>Flanco activo de la interrupción externa.  <b>1</b>= Flanco ascendente.  <b>0</b>= Flanco descendente.</p>																											
<b>RBPUP:</b>	<p>Resistencias Pull-up Puerta B.  <b>1</b>= Desactivadas.  <b>0</b>= Activadas.</p>																											

### *Interrupciones y sus Principales Registros*

Para el Pic16F877A que se utiliza en este proyecto tiene 14 fuentes de interrupción, cuando se produce alguna interrupción el valor del PC se guarda en la pila y se carga en el PC el valor 0004h que es el vector de interrupción y en donde se guarda la parte del código que debe realizar al provocarse la interrupción.

Las fuentes de interrupción se nombran a continuación en el Pic16f877A:

1. Por desbordamiento del temporizador TMR0.
2. Activación de la patita de interrupción RP0/INT.
3. Cambio de estado de las 4 patitas de más peso de la Puerta B.
4. Finalización de escritura de un byte en la EEPROM.
5. Por desbordamiento del Timer1.
6. Por desbordamiento del Timer2.
7. Captura o comparación en el módulo CCP1.

8. Captura o comparación en el módulo CCP2.
9. Transferencia en la SSP (Puerta serie sincrónica).
10. Colisión de bus en el bus en la SSP (Puerta serie sincrónica).
11. Fin de la transmisión en el USART.
12. Fin de la recepción en el USART.
13. Fin de conversión en el convertidor A/D.
14. Transferencia en la PSP (Puerta Paralela Esclava).

Para que cada una de estas interrupciones se produzca hay que configurar ciertos registros, el principal es el registro INTCON que además sirve para observar la causa de la interrupción. Los registros para controlar los periféricos especiales son:

- PIE1
- PIR1
- PIE2
- PIR2

En el registro INTCON hay un bit GIE (INTCON<7>) Permiso Global de interrupciones, el cual debe ser habilitado para que se produzcan cualquiera de las 14 interrupciones, pero cada interrupción tiene su bit de habilitación en los registros anteriormente mencionados. Para regresar del código que ejecuta después de la interrupción se ejecuta la instrucción RETFIE. Existe además en el registro INTCON el bit de habilitación de interrupción de los periféricos PEIE que permite que se produzcan las interrupciones de los periféricos que no se controlan con el INTCON. Luego cuando se producen las interrupciones por lo general los señalizadores hay que limpiarlos por software antes de que se produzca otra interrupción. Si GIE es limpiado, se ignorara toda interrupción que se produzca. A continuación se muestra los bits de los registros que manejan las interrupciones y en las tablas VIII y IX se encuentran la función de los Registros INTCON y PIE1:

### REGISTRO INTCON

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GIE	PEIE <sup>(3)</sup>	TOIE	INTE <sup>(2)</sup>	RBIE <sup>(1, 2)</sup>	TOIF	INTF <sup>(2)</sup>	RBIF <sup>(1, 2)</sup>
bit 7							bit 0

Tabla VIII

Función de los bits del registro INTCON.

<b>RBIF:</b>	Señalizador de cambio en RB4-RB7.
<b>INTF:</b>	Señalizador por interrupción en RB0/INT.
<b>TOIF:</b>	Señalizador de desbordamiento en TMR0.
<b>RBIE:</b>	Permiso de interrupción por cambio en RB4-RB7.
<b>INTE:</b>	Permiso de interrupción en la patita RB0/INT.
<b>TOIE:</b>	Permiso de interrupción del temporizador TMR0.
<b>PEIE:</b>	Permiso de interrupción de los Periféricos que no se controlan con INTCON.
<b>GIE:</b>	Permiso global de interrupciones.

**REGISTRO PIE1**

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 <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit 7							bit 0

Tabla IX

## Función de los bits del registro PIE1

<b>TMR1IE:</b>	Permiso de interrupción por desbordamiento del TMR1.
<b>TMR2IE:</b>	Permiso de interrupción por desbordamiento del TMR2.
<b>CCP1IE:</b>	Permiso de interrupción para el módulo CCP1.
<b>SSPIE:</b>	Permiso de interrupción para la Puerta serie sincrónica (PSP).
<b>TXIE:</b>	Permiso de interrupción para el transmisor del USART cuando el buffer se vacía.
<b>RCIE:</b>	Permiso de interrupción para el receptor del USART cuando el buffer se llena.
<b>ADIE:</b>	Permiso de interrupción para el convertidor A/D al fin de la conversión.
<b>PSPIE:</b>	Permiso de interrupción para puerta paralela esclava (PSP) al leer/escribir.

El registro PIE1 ocupa la dirección 8Ch y para que cumplan su función los bits PIE1 es necesario que PEIE =1 en INTCON<6>. En la tabla X se muestra la función de los bits del registro PIE2.

**REGISTRO PIE2**

U-0	R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0
—	Reserved	—	EEIE	BCLIE	—	—	CCP2IE
bit 7							bit 0



Tabla X

Función de los bits del registro PIE2.

<b>EEIE:</b>	Permiso de interrupción por fin de Escritura en la EEPROM.
<b>BCLIE:</b>	Permiso de interrupción por colisión en el bus SSP cuando dos o más maestros tratan de transmitir al mismo tiempo.
<b>CCP2IE:</b>	Permiso de interrupción en el módulo CCP2.

Las posiciones con trama oscura no están implementadas y se leen siempre como cero. El registro PIE2 ocupa la posición 8Dh.

**Registro PIR1**

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

**Registro PIR2**

U-0	R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0
—	Reserved	—	EEIF	BCLIF	—	—	CCP2IF
bit 7							bit 0

Los registros PIR1 y PIR2 son registros que contienen los señalizadores de las interrupciones de los periféricos internos anteriormente mencionados en PIE1 y PIE2.

La figura 4.9 siguiente muestra la lógica de las interrupciones:

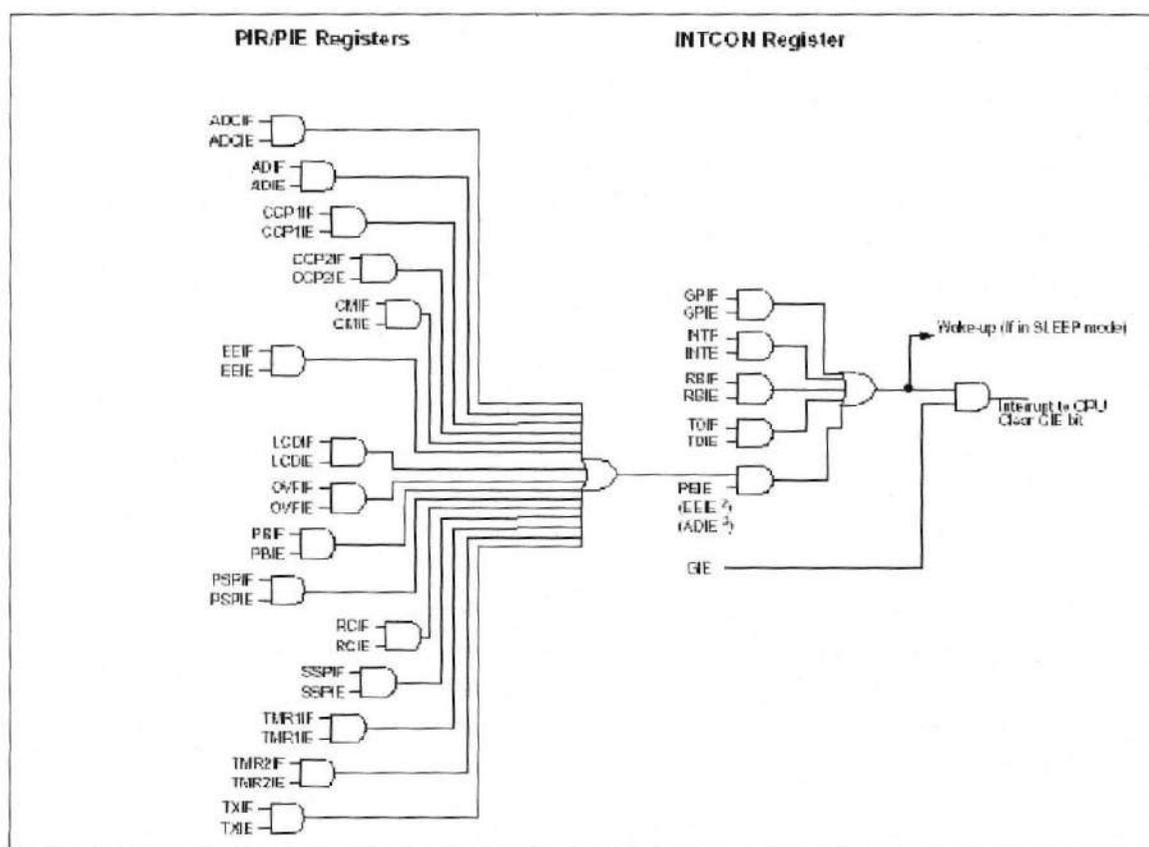


Figura 4.9.-Diagrama de Conexiones de las fuentes de Interrupciones...  
Memoria de datos EEPROM y memoria de código FLASH

En el sistema de monitoreo de baterías realizado se usa la memoria de datos EEPROM, se realizan las operaciones de Lectura y Escritura por lo que vamos a explicar el funcionamiento de la memoria de datos EEPROM y las implementaciones que se pueden realizar en este modelo de PIC 16F877A. La EEPROM y la memoria de código Flash se pueden leer y escribir durante la operación normal del rango completo de Vdd que se verán en las especificaciones eléctricas. Estas operaciones utiliza un simple byte para la EEPROM y una palabra en la memoria de código FLASH. Acceder a la memoria FLASH permite el llamado "checksum" para control de errores. Los valores escritos en la memoria de código no necesitan ser instrucciones válidas. Sin embargo, números mayores a 14 bits pueden ser guardados en memoria para usos como calibración de parámetros, números de serie, etc. Luego al ejecutarse una localidad de la memoria de código que tiene almacenado un dato la operación que realiza es un Nop (ninguna operación, pero ocupa un ciclo de instrucción).

La escritura de un byte o palabra automáticamente borra dicha localidad y escribe el nuevo valor, por lo que escribiendo la EEPROM no afectará la operación del dispositivo, pero al escribir en la FLASH detendrá la ejecución de instrucciones hasta que la escritura sea completada así la memoria de código FLASH no puede ser accedida durante la escritura. Durante la escritura el oscilador sigue corriendo, los periféricos siguen funcionando y si algún interrupción se produce debe esperar hasta que termine la escritura. Con la Lectura/Escritura de la FLASH significa que un programa puede ser dinámico y grabar información en la Flash sin necesidad de un grabador externo, por lo que el programa podría grabar datos y luego actuar de acuerdo a lo que ya ha grabado como un proceso de aprendizaje, son los principios de sistemas inteligentes.

Para realizar estas operaciones no es suficiente con un solo registro para la dirección que alcanza los 13 bits, y lo mismo sucede para el dato, que tiene una longitud de 14 bits, por lo tanto se usa el registro EEADR concatenado con EEADRH, que contiene los 5 bits de más peso de la dirección, mientras que para el dato se utiliza el registro EEDATAH que concatena con el EEDATA, y que contiene los 6 bits más significativos de la palabra leída o a escribir en la

FLASH. Los registros EEADRH y EEDATAH no se usan para trabajar con la EEPROM. Para controlar la Lectura/Escritura de la EEPROM y la FLASH hay dos registros EECON1 y EECON2.

### *Registros EECON1 y EECON2*

El EECON1 es un registro de control mientras que EECON2 es un registro que no está implementado físicamente pero es usado exclusivamente para la operación de la escritura, que tiene una duración aproximada de 2 milisegundos para prevenir alguna inadvertida escritura. El bit EEPGD (EECON1<7>) determina si el acceso es a la EEPROM o a la FLASH. Para operaciones de lectura se usa un bit adicional RD (EECON1<0>) el cual inicia la lectura de la localidad de memoria, una vez RD es uno la localidad de memoria estará en el registro de datos, este bit no es limpiado por software sino que se pone a cero automáticamente al final de la operación de lectura. Para lecturas de la EEPROM, el dato se encuentra en EEDATA después de un ciclo de instrucción de que se estableció RD a uno y para lecturas de la FLASH el dato estará en EEDATAH: EEDATA después de 2 ciclos de instrucción de que RD se estableció a uno. Para operaciones de escritura tiene dos bits de control WR y



Tabla XI

Función de los bits del registro EECON1.

<b>EEPGD:</b>	Selecciona el acceso a la FLASH igual a 1 o a la EEPROM igual a cero.
<b>WRERR:</b>	Señalizador de error en la escritura.
<b>WREN:</b>	Permiso de la escritura.
<b>WR:</b>	Hay que establecer a uno para iniciar la escritura.
<b>RD:</b>	Hay que poner a uno para iniciar la lectura.

*Lectura de la EEPROM de datos*

Los pasos utilizados en el Firmware del Pic para la lectura de la EEPROM son:

- 1) Escribir la dirección en EEDATA. Asegurar que la dirección no es tan larga que el tamaño de la memoria.
- 2) Establecer el bit EEPGD a cero para que la lectura sea a la EEPROM.
- 3) Poner a 1 RD para iniciar la lectura.
- 4) Leer el dato del registro EEDATA.

El siguiente ejemplo muestra la lectura de la EEPROM:

BPF	STATUS, RP1	;
BPF	STATUS, RP0	;Bank 2
MOVF	ADDR, W	;Write address
MOVWF	EEADR	;to read from
BPF	STATUS, RP0	;Bank 2
BPF	EECON1, EEPGD	;Point to Data memory
BPF	EECON1, RD	;Start read operation
BPF	STATUS, RP0	;Bank 2
MOVF	EEDATA, W	;W = EEDATA

#### *Escritura de la EEPROM de datos*

Los pasos utilizados en el Firmware del Pic para la escritura de la EEPROM son:

- 1) Verificar si ninguna operación de escritura se está llevando a cabo con el bit WR
- 2) Escribir la dirección en EEADR, pero hay que asegurarse que la dirección no es más larga que el tamaño de la memoria del dispositivo.
- 3) Escribir los 8 bits del dato en EEDATA.
- 4) Poner EEPGD a cero para trabajar en la EEPROM de datos.

- 5) Establecer WREN a uno para habilitar las operaciones de Lectura/Escritura.
- 6) Deshabilitar las interrupciones.
- 7) Hay que ejecutar las siguiente 5 instrucciones:
  - Escribir 55h al registro EECON2.
  - Escribir AAh al registro EECON2.
  - Poner WR a uno.
- 8) Habilitar las interrupciones deshabilitadas.
- 9) Limpiar el WREN para deshabilitar las operaciones en la EEPROM.
- 10) Al término de la escritura de la EEPROM WR se pone a cero mientras que el señalizador EEIF es uno, el cual debe ser borrado por software.

El siguiente ejemplo muestra la escritura en la EEPROM:

```

RSP STATUS, RFI ;
RSP STATUS, RFO ;Bank 3
RSPBC EEPROM, WR ;Wait for
GOTO $-1 ;write to finish
RSP STATUS, RFO ;Bank 2
MOVW ADDR, W ;Address to
MOVWF EEAR ;write to
MOVW VALUE, W ;Data to
MOVWF EEADATA ;write
RSP STATUS, RFO ;Bank 1
RSP EEPROM, REND ;Point to Data memory
RSP EEPROM, WREN ;Enable writes
RSP INTOCN, GIE ;if already enabled,
;otherwise discard
MOVW 0x55 ;Write 55h to
MOVWF EEPROM ;EEPROM
MOVW 0xAA ;Write AAh to
MOVWF EEPROM ;EEPROM
RSP EEPROM, WR ;start write operation
;only enable interrupts
RSP INTOCN, GIE ;if using interrupts,
;otherwise discard
RSP EEPROM, WREN ;Disable writes

```

### *Lectura de la memoria FLASH*

Los pasos para la lectura de la memoria FLASH:

- 1) Escribir la dirección en EEADRH: EEADR asegurando que la dirección no sea más larga que el tamaño de la memoria del dispositivo.
- 2) Establecer el acceso a la memoria FLASH poniendo EEPGD a uno.
- 3) Poner RD a uno para habilitar la operación de la lectura.

- 4) Ejecute la instrucción Nop dos veces para permitir al Pic para termine la lectura y obtener el dato.
- 5) Leer el dato de EEDATH: EEDATA.

El siguiente ejemplo muestra la lectura de la FLASH:

```

NOP    STATUS, RPI    ;
NOP    STATUS, RPO    ;Bank 2
MOVWF  ADDR1, W      ;Write the
MOVWF  ADDR2         ;address bytes
MOVWF  ADDR3, W      ;for the desired
MOVWF  ADDR4         ;address to read
NOP    STATUS, RPO    ;Bank 1
NOP    RECON1, REPC1  ;Point to Program Memory
NOP    RECON1, RD     ;start read operation
NOP                                     ;Required two NOPs
NOP
NOP    STATUS, RPO    ;Bank 2
MOVWF  EEDATA, W     ;DATA1 = EEDATA
MOVWF  DATA1        ;
MOVWF  EEDATH, W     ;DATA1 = EEDATH
MOVWF  DATA1        ;

```



CIB-ESPOL

### *Escritura de la memoria FLASH*

Los pasos para la escritura en la memoria FLASH son:

- 1) Escribir la dirección en EEADRH: EEADR asegurando que la dirección no sea más larga que el tamaño de la memoria del dispositivo.
- 2) Escribir el dato de 14 bits en el registro EEDATH: EEDATA.
- 3) Establecer el acceso a la memoria FLASH poniendo EEPGD a uno.
- 4) Establecer a uno el bit WREN para habilitar las operaciones de Lectura/Escritura.
- 5) Deshabilitar las interrupciones.
- 6) Hay que ejecutar las siguientes 5 instrucciones:
  - Escribir 55h al registro EECON2.
  - Escribir AAh al registro EECON2.
  - Poner WR a uno.
- 7) Ejecute dos instrucciones Nop para permitir al Pic que termine la escritura.
- 8) Habilitar las interrupciones.
- 9) WREN a cero para deshabilitar el trabajo con la memoria FLASH.

El siguiente ejemplo muestra la escritura en la FLASH:

```

BCF STATUS, RP1 ;
BCF STATUS, RP0 ;Bank 1
MOVF ADDR1, W ;Write address
MOVWF ADDR ;of desired
MOVWF ADDRH, W ;program memory
MOVWF ADDRHH ;location
MOVF VAL1, W ;Write value to
MOVWF VAL1A ;program at
MOVF VAL1RH, W ;desired memory
MOVWF VAL1HH ;location
BCF STATUS, RP0 ;Bank 0
BCF EECOM1, EEPED ;Point to Program memory
BCF EECOM1, WHEN ;Enable writes
BCF INTCON, GIE ;Only disable interrupts
;if already enabled,
;otherwise discard
MOVLW 0x55 ;Write 55h to
MOVWF EECOM2 ;EECOM2
MOVLW 0xAA ;Write AAh to
MOVWF EECOM3 ;EECOM3
BCF EECOM1, WR ;Start write operation
NOP ;Two NOPs to allow micro
NOP ;to setup for write
BCF INTCON, GIE ;Only enable interrupts
;if using interrupts,
;otherwise discard
BCF EECOM1, WHEN ;Disable writes

```

#### 4.1.5 PUERTAS DE ENTRADA/SALIDA.

El Pic16877A usado en este proyecto tiene 5 puertas A, B, C, D, E dichas líneas son bidireccionales y multifuncionales dependiendo de la programación que se le de al dispositivo. La dirección de las Puertas E/S es controlada por un registro llamado TRIS cada línea de la puerta puede ser configurada como

entrada o salida, un "1" configura dicho pin como entrada mientras que un "0" configura el pin como salida.

Otra característica es que el dato es mantenido o guardado en una especie de memoria en la salida cuando el pin de la puerta es configurado como salida. Cuando la puerta es leída, el dispositivo lee el nivel presente en el pin de E/S no el dato guardado en la memoria. Por lo que hay que tener cuidado en operaciones de lectura-modificación-escritura en las puertas y en los cambios de dirección de un pin de entrada a salida.

#### *Puerta A y Registro TRISA*

Dispone de 6 líneas (RA0-RA5), son bidireccionales y como ya se dijo con el registro TRISA se configura cada pin o línea como entrada o salida. Un "1" configura a una línea como entrada mientras que un "0" como salida y el contenido de la memoria llamada en inglés "Data Latch" que es un flip-flop se aplica a dicha línea. Al leer la Puerta A se lee el estado de los pines. La escritura es una operación de lectura-modificación-escritura, es decir que primero lee las patitas, modifica su valor y finalmente se escribe en el Data

Match. Los pines que tienen multiplexados sus funciones son RA0/AN0, RA1/AN1 que además de actuar como E/S digitales, sirve como canales 0, 1 por donde se puede aplicar las señales analógicas del convertidor A/D. Los pines RA2/AN2/ $V_{ref-}$  como canal 2 o voltaje de referencia negativo del módulo A/D y RA3/AN3/ $V_{ref+}$  como canal 3 o como voltaje de referencia para el módulo A/D. El pin RA4/TOCK1 actúa como E/S digital y como entrada de la señal de reloj para el TMR0. El pin RA5/AN4/SS# funciona como E/S digital, canal 4 para el módulo A/D y selección del modo esclavo cuando se trabaja con la comunicación serie sincrónica. En la tabla 4.12 se describe la función de cada bit de la Puerta A.

Tabla XII

Funciones de cada pin de la Puerta A.

Nombre	Bit#	Buffer	Función
RA0/AN0	bit 0	TTL	E/S o Canal analógica.
RA1/AN1	bit 1	TTL	E/S o Canal analógico.
RA2/AN2/ $V_{ref-}$	bit 2	TTL	E/S o Canal analógico o $V_{ref-}$ .
RA3/AN3/ $V_{ref+}$	bit 3	TTL	E/S o Canal analógico o $V_{ref+}$ .
RA4/ TOCK1	bit 4	Schmitt Trigger	E/S o Entrada reloj externo para TMR0.
RA5/AN4/ SS#	bit 5	TTL	E/S o Canal analógico o entrada selección esclavo SSP.

En la figura 4.10, se muestra el esquema de conexión de los pines RA0:RA3 y RA5:

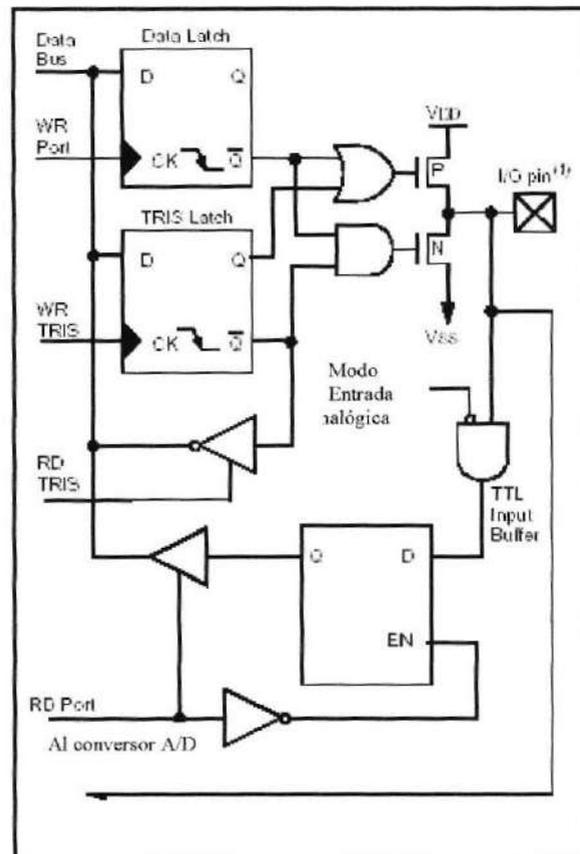


Figura 4.10.- Esquema de Conexiones de la Puerta A.

La figura 4.11 muestra el diagrama de conexiones de RA4/TOCK1:

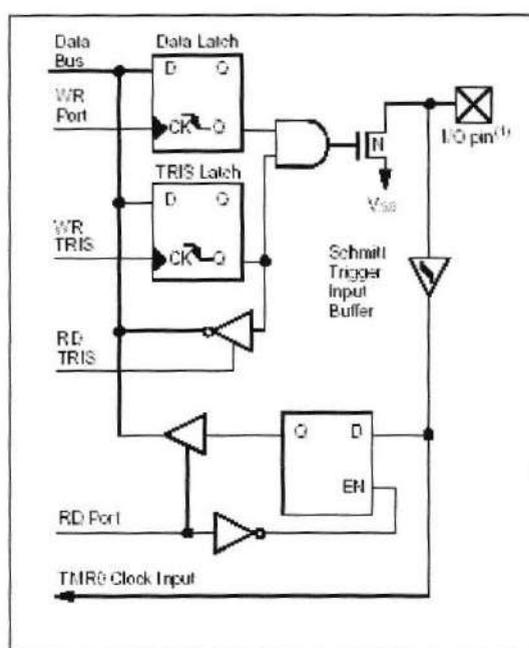


Figura 4.11.- Esquema de Conexiones de la Puerta A.

Para utilizar las líneas de la puerta A como digitales hay que configurar el registro ADON1 del convertidor A/D con el valor adecuado que más adelante veremos.

A continuación pongo un ejemplo de la configuración de las líneas de la Puerta A:

```

BCF STATUS, RPO ;
BCF STATUS, RPI ; Bank0
CLR PORTA ; Initialize PORTA by
; clearing output
; data latches

BCF STATUS, RPO ; Select Bank 1
MOVW 0x00 ; Configure all pins
MOVWF ADCON1 ; as digital inputs
MOVW 0x0F ; Value used to
; initialize data
; direction
MOVWF TRISA ; Set RA0:RA5 as inputs
; RA6:RA7 as outputs
; TRISA0:TRISA7 always
; read as '0'.

```

### *Puerta B y Registro TRISB*

Dispone de 8 líneas bidireccionales cuya función se elige configurando el registro TRISB. Tres pines de la Puerta B son multiplexadas para la Programación de voltaje bajo RB3/PGM, RB6/PGD y RB7/PGD y se introduce un voltaje de 5V por MCLR#/Vpp el pin RB3/PGM hay que conectarlo a nivel alto y como la programación siempre se realiza sincrónicamente en serie, por el pin RB6/PGC se introducen los impulsos del reloj y por la RB7/PGD los bits de datos en serie. También hay una programación de voltaje alto, este se introduce

MCLR#/Vpp y está comprendido entre 12 y 14 V. Cada pin de la Puerta B tiene una resistencia interna "Pull-up" conectada a la alimentación, que es un transistor CMOS tipo P, dicha resistencia "Pull-Up" se conecta a la alimentación cuando un bit RBPU# del registro OPTION tiene el valor de cero siempre que la línea esté configurada como salida. Las resistencias "Pull-Up" se desconectan cuando se produce un POR (Conexión a la alimentación). Cuatro líneas de la Puerta B producen una interrupción cuando existe un cambio en las líneas RB7:RB4. Dichas líneas deben ser configuradas como entradas. Las entradas en RB7:RB4 son comparadas con la anterior y si no coinciden se genera la interrupción siempre que lo autorice el bit de permiso, que se encuentra en INTCON, esta situación puede ser advertido por el señalizador RBIF. El pin RB0/INT puede funcionar como E/S digital y como petición de interrupción externa pero poniendo a uno el bit de permiso del registro INTCON. La Tabla XIII describe las funciones de los bits de la Puerta B.

Tabla XIII  
Funciones de cada pin de la Puerta B

Nombre	Bit#	Buffer	Función
RB0/INT	bit 0	TTL/ST	E/S o Interrupción externa o resistencia "Pull-Up".
RB1	bit 1	TTL	E/S o resistencia "Pull-Up".
RB2	bit 2	TTL	E/S o resistencia "Pull-Up".
RB3/PGM	bit 3	TTL	E/S o Programación LVP o resistencia "Pull-Up".
RB4	bit 4	TTL	E/S o resistencia "Pull-Up".
RB5	bit 5	TTL	E/S o resistencia "Pull-Up".
RB6/PGC	bit 6	TTL/ST	E/S o resistencia "Pull-Up". Entrada serial de reloj.
RB7/PGD	bit 7	TTL/ST	E/S o resistencia "Pull-Up". Entrada serial de Datos.

La figura 4.12 muestra el diagrama de conexiones de la Puerta B, pines RB3:RB0:





La figura 4.13 muestra el diagrama de la Puerta B, pines RB7:RB4:

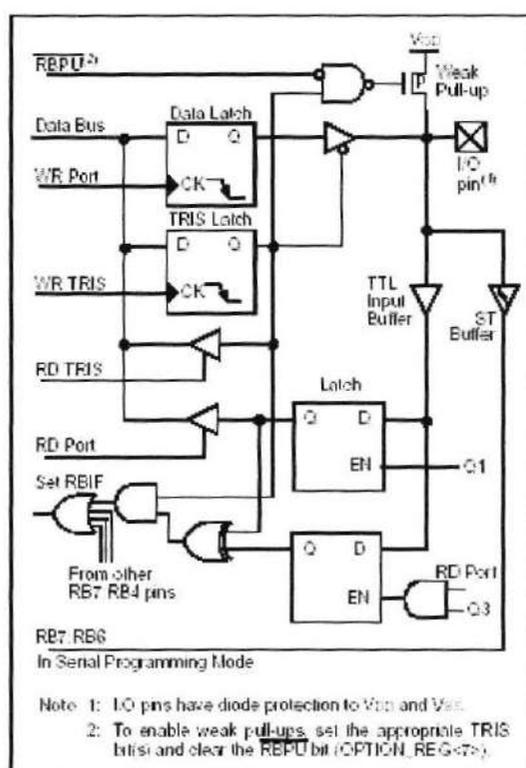


Figura 4.13.- Diagrama de Conexiones de la Puerta B.

El siguiente ejemplo muestra una inicialización de la Puerta B.

```

CLRF STATUS      ; Bank0
CLRF PORTE       ; Initialize PORTE by clearing output
                  ; data latches
BSF STATUS, RP0  ; Select Bank1
MOVLW 0xCF       ; Value used to initialize data direction
MOVWF TRISE      ; PORTE<3:0> = inputs, PORTE<5:4> = outputs
                  ; PORTE<7:6> = inputs

```

### *Puerta C Y Registro TRISC*

Dispone de 8 líneas bidireccionales en el que el sentido de las líneas se configura en el Registro TRISC. La Puerta C tiene todos sus pines multiplexados con diferentes funciones, además todos tienen un schmitt trigger cuando funcionan como entrada. En la tabla XIV se describen las funciones de cada bit de la puerta C.

Tabla XIV

Funciones de cada pin de la Puerta C.

Nombre	Función
RC0/T1OSO/T1CK1	Actúa como E/S digital, salida del Timer 1 o entrada de impulsos para el Timer 1.
RC1/T1OSI/CCP2	E/S digital/ Entrada al oscilador del Timer 1 /entrada del módulo de Captura 2; Salida del comparador 2; Salida del PWM.
RC2/CCP1	E/S digital/ Entrada Captura 1, Salida del comparador 1; Salida PWM 1.
RC3/SCK/SCL	E/S digital/Señal de reloj en modo SPI/Señal de reloj en modo I2C.
RC4/SDI/SDA	E/S digital/Entrada de datos en modo SPI/Línea de datos en modo I2C.
RC5/SDO	E/S digital/Salida de datos en modo SPI.
RC6/TX/CK	E/S digital/Línea de transmisión en USART/Señal de

	reloj síncrona en transmisión serie.
RC7/RX/DT	E/S digital/Línea de recepción del USART/Línea de datos en transmisión serie síncrona.

La figura 4.14 muestra el diagrama de conexiones de la Puerta C, pines RC7:RC5 y RC2:RC0.

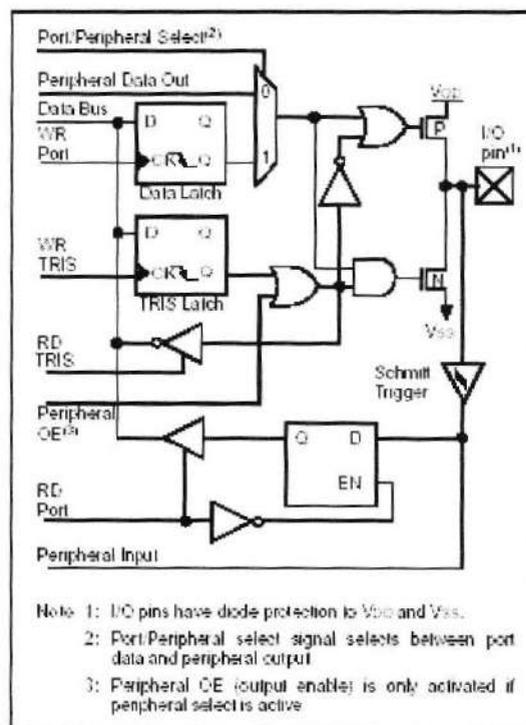


Figura 4.14.- Diagrama de Conexiones de la Puerta C.

La figura 4.15 muestra el diagrama de conexiones de la Puerta C, pines RC4:RC3:

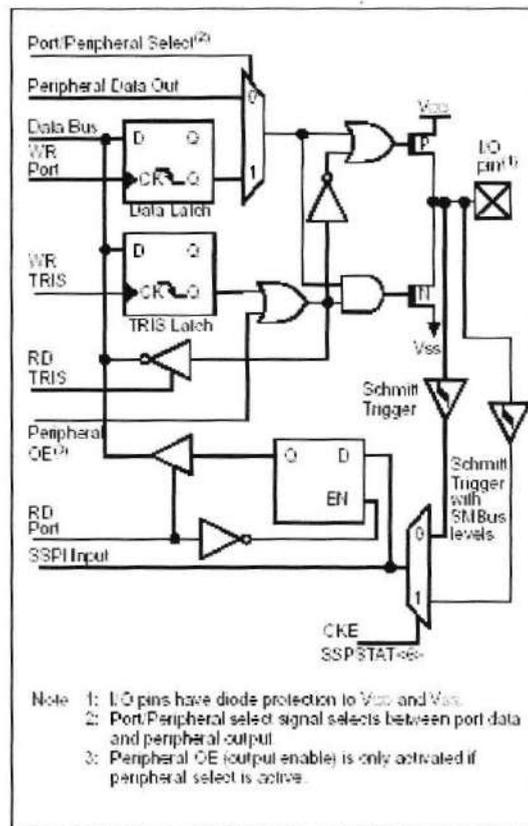


Figura 4.15.- Diagrama de Conexiones de la Puerta C.

Como se puede ver en el gráfico cuando se utiliza periféricos como el I2C, la puerta C puede ser configurada con niveles del bus I2C normales o con niveles SMBus pero usando el bit CKE (SSPSTAT<6>).

#### *Puerta D y Registro TRISD.-*

La puerta D tiene 8 líneas bidireccionales, todas ellas tienen un buffer schmitt trigger cuando funcionan como entrada. Cada línea debe ser configurada independientemente como entrada o salida. La Puerta D puede trabajar como una puerta paralela esclava para comunicaciones en paralelo con otros elementos del sistema, para lo que hay que configurar adecuadamente el bit PSPMODE (TRISE<4>). La Tabla XV muestra las funciones de cada bit de la Puerta D.

Tabla XV

Funciones de cada pin de la Puerta D.

Nombre	Bit#	Buffer	Función
RD0/PSP0	bit 0	ST/TTL	E/S digitales o Puerta paralela esclava (PSP).
RD1/PSP1	bit 1	ST/TTL	E/S digitales o Puerta paralela esclava (PSP).
RD2/PSP2	bit 2	ST/TTL	E/S digitales o Puerta paralela esclava (PSP).
RD3/PSP3	bit 3	ST/TTL	E/S digitales o Puerta paralela esclava (PSP).
RD4/PSP4	bit 4	ST/TTL	E/S digitales o Puerta paralela esclava (PSP).
RD5/PSP5	bit 5	ST/TTL	E/S digitales o Puerta paralela esclava (PSP).
RD6/PSP6	bit 6	ST/TTL	E/S digitales o Puerta paralela esclava (PSP).
RD7/PSP7	bit 7	ST/TTL	E/S digitales o Puerta paralela esclava (PSP).

La figura 4.16 muestra el diagrama de conexiones de la Puerta D:

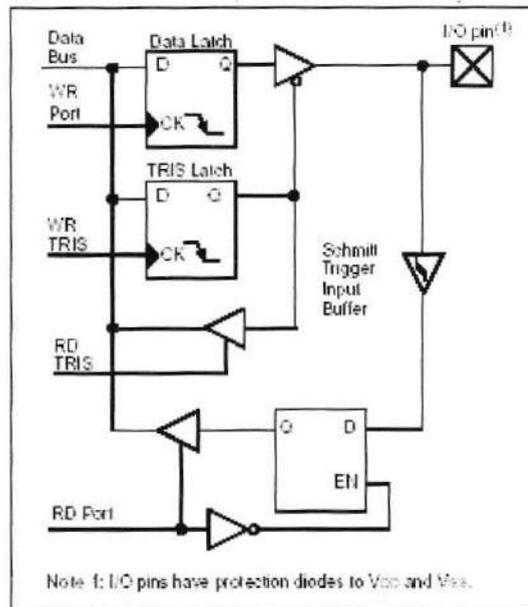


Figura 4.16.- Diagrama de Conexiones de la Puerta D.

#### *Puerta E y Registro TRISE.-*

Dispone de tres bits, los que se pueden configurar como entradas o salidas con los 3 bits de menos peso del registro TRISE. Cuando el bit PSPMODE se pone a uno, las líneas del puerto E se vuelven entradas controladas por la PSP, la cual se usan para determinar si la operación es de lectura, escritura y de permiso de funcionamiento RD#, WR# y CS#.

Las puerta E puede ser utilizada también como E/S digitales o Analógicas configurando el registro ADCON1. La Tabla XVI muestra la función de cada bit de la Puerta E.

Tabla XVI

Funciones de cada pin de la Puerta E.

Nombre	Bit#	Buffer	Función
RE0/RD/AN 5	bit 0	ST/TTL	E/S digitales o control de lectura de la Puerta paralela esclava (PSP) RD o canal analógico...
RE1/WR/AN 6	bit 1	ST/TTL	E/S digitales o control de escritura de la Puerta paralela esclava (PSP) WR o canal analógico.
RE2/CS/AN 7	bit 2	ST/TTL	E/S digitales o selección del dispositivo de la Puerta paralela esclava (PSP) CS o canal analógico.



El tipo de conversión que usa es el de aproximaciones sucesivas. La resolución de la conversión depende de un voltaje de referencia, vía software se determina que voltaje de referencia se usa, referencias externas o la referencia del Pic (Vdd, Pin 11 y 32).

La resolución del convertidor está dada por la Ecuación 4.1:

$$\text{Resolución} = \frac{(V_{ref} - V_{ref-})}{1024} = \frac{V_{ref}}{1024}$$

Ecuación 4.1.-

El módulo de conversión A/D, es el único módulo del Pic que puede trabajar en el modo SLEEP o de reposo, para ello deberá conectar el reloj del conversor al oscilador RC interno.



### *Registros de Trabajo del Conversor*

El módulo conversor A/D utiliza 4 registros de trabajo:

- Resultado de la conversión más significativa (ADRESH).
- Resultado de la conversión menos significativa (ADRESL)
- Registro de control 0 (ADCON0)
- Registro de control 1 (ADCON1)

El registro de control ADCON0, controla la operación del módulo A/D y el registro de control ADCON1 configura las funciones de los pines de las Puerta A. En la Tabla XVII y la Tabla XVIII se indica la función de cada bit del registro ADCON0.

#### **REGISTRO ADCON0**

RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	U-0	RW-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON
bit 7							bit 0

Tabla XVII

## Función de los bits del Registro ADCON0

<b>ADCS1: ADCS0</b>	Sirve para seleccionar la frecuencia del reloj para la conversión.: 00 $F_{osc}/2$ , 01 $F_{osc}/8$ , 10 $F_{osc}/32$ , 11 $F_{RC}$ (Oscilador interno RC).
<b>CHS2: CHS0:</b>	Sirven para seleccionar el canal por donde se aplica la señal analógica a convertir.
<b>GO/DONE:</b>	Sirve para iniciar la conversión, cuando se pone este bit a uno se inicia la conversión, luego automáticamente se pone a cero al termino de la misma.
<b>ADON:</b>	Sirve para habilitar el módulo A/D.

## REGISTRO ADCON1

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
bit 7							bit 0

Tabla XVIII

Funciones de los bits del registro ADCON1.

<b>ADFM:</b>	<p>Sirve para justificar el resultado de la conversión.</p> <ul style="list-style-type: none"> <li>▪ 1= justificación a la derecha, los 6 bits más significativos de ADRESH son leídos como cero.</li> <li>▪ 0= justificación a la izquierda, los 6 bits menos significativos ADRESL son leídos como ceros.</li> </ul>																																																																																																																																																																																																
<b>PCFG3:PCFG0</b>	<p>Sirve para configurar los pines de los canales al convertor como analógicas o como E/S digitales, en el cuadro se puede observar los valores que deben tomar estos bits.</p> <table border="1" data-bbox="429 928 1346 1474"> <thead> <tr> <th>PCFG3: PCFG0</th> <th>AN7<sup>(1)</sup> RE2</th> <th>AN6<sup>(1)</sup> RE1</th> <th>AN5<sup>(1)</sup> RE0</th> <th>AN4 RA5</th> <th>AN3 RA3</th> <th>AN2 RA2</th> <th>AN1 RA1</th> <th>AN0 RA0</th> <th>VREF+</th> <th>VREF-</th> <th>Chan/ Refs<sup>(2)</sup></th> </tr> </thead> <tbody> <tr><td>0000</td><td>A</td><td>A</td><td>A</td><td>A</td><td>A</td><td>A</td><td>A</td><td>A</td><td>VDD</td><td>VSS</td><td>8/0</td></tr> <tr><td>0001</td><td>A</td><td>A</td><td>A</td><td>A</td><td>VREF+</td><td>A</td><td>A</td><td>A</td><td>RA3</td><td>VSS</td><td>7/1</td></tr> <tr><td>0010</td><td>D</td><td>D</td><td>D</td><td>A</td><td>A</td><td>A</td><td>A</td><td>A</td><td>VDD</td><td>VSS</td><td>5/0</td></tr> <tr><td>0011</td><td>D</td><td>D</td><td>D</td><td>A</td><td>VREF+</td><td>A</td><td>A</td><td>A</td><td>RA3</td><td>VSS</td><td>4/1</td></tr> <tr><td>0100</td><td>D</td><td>D</td><td>D</td><td>D</td><td>A</td><td>D</td><td>A</td><td>A</td><td>VDD</td><td>VSS</td><td>3/0</td></tr> <tr><td>0101</td><td>D</td><td>D</td><td>D</td><td>D</td><td>VREF+</td><td>D</td><td>A</td><td>A</td><td>RA3</td><td>VSS</td><td>2/1</td></tr> <tr><td>011x</td><td>D</td><td>D</td><td>D</td><td>D</td><td>D</td><td>D</td><td>D</td><td>D</td><td>VDD</td><td>VSS</td><td>0/0</td></tr> <tr><td>1000</td><td>A</td><td>A</td><td>A</td><td>A</td><td>VREF+</td><td>VREF-</td><td>A</td><td>A</td><td>RA3</td><td>RA2</td><td>6/2</td></tr> <tr><td>1001</td><td>D</td><td>D</td><td>A</td><td>A</td><td>A</td><td>A</td><td>A</td><td>A</td><td>VDD</td><td>VSS</td><td>0/0</td></tr> <tr><td>1010</td><td>D</td><td>D</td><td>A</td><td>A</td><td>VREF+</td><td>A</td><td>A</td><td>A</td><td>RA3</td><td>VSS</td><td>5/1</td></tr> <tr><td>1011</td><td>D</td><td>D</td><td>A</td><td>A</td><td>VREF+</td><td>VREF-</td><td>A</td><td>A</td><td>RA3</td><td>RA2</td><td>4/2</td></tr> <tr><td>1100</td><td>D</td><td>D</td><td>D</td><td>A</td><td>VREF+</td><td>VREF-</td><td>A</td><td>A</td><td>RA3</td><td>RA2</td><td>3/2</td></tr> <tr><td>1101</td><td>D</td><td>D</td><td>D</td><td>D</td><td>VREF+</td><td>VREF-</td><td>A</td><td>A</td><td>RA3</td><td>RA2</td><td>2/2</td></tr> <tr><td>1110</td><td>D</td><td>D</td><td>D</td><td>D</td><td>D</td><td>D</td><td>D</td><td>A</td><td>VDD</td><td>VSS</td><td>1/0</td></tr> <tr><td>1111</td><td>D</td><td>D</td><td>D</td><td>D</td><td>VREF+</td><td>VREF-</td><td>D</td><td>A</td><td>RA3</td><td>RA2</td><td>1/2</td></tr> </tbody> </table>	PCFG3: PCFG0	AN7 <sup>(1)</sup> RE2	AN6 <sup>(1)</sup> RE1	AN5 <sup>(1)</sup> RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	Chan/ Refs <sup>(2)</sup>	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	0/0	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
PCFG3: PCFG0	AN7 <sup>(1)</sup> RE2	AN6 <sup>(1)</sup> RE1	AN5 <sup>(1)</sup> RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	Chan/ Refs <sup>(2)</sup>																																																																																																																																																																																						
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	0/0																																																																																																																																																																																						
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																																																																																																																																																																																						

La conversión se inicia poniendo a 1 el bit GO/DONE, al terminar la conversión se pone a cero y el resultado se guardará en ADRESH: ADRESL, según como

se haya configurado la justificación con el bit ADFM (ADCON1<7>), en la figura 4.18 se puede observar la justificación en los registros ADRESH: ADRESL.

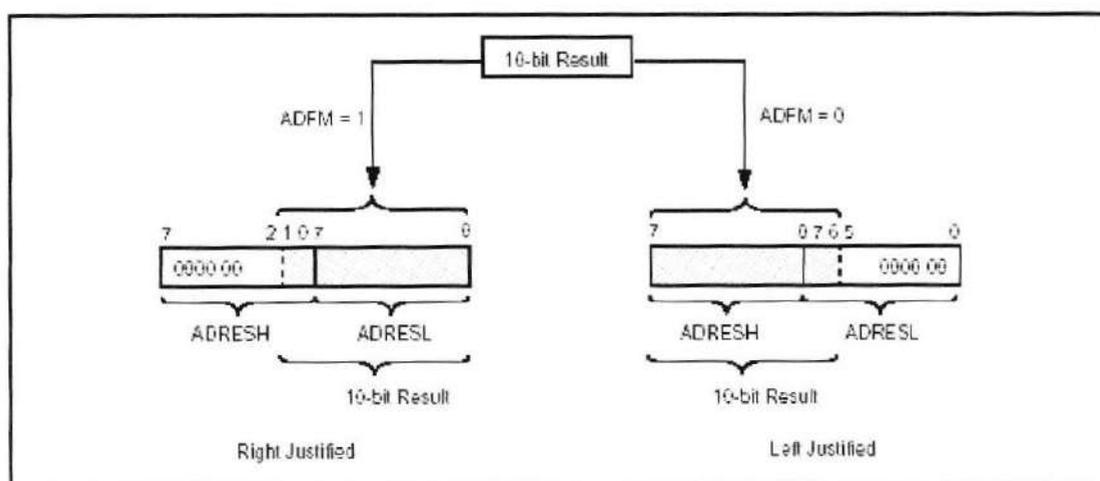


Figura 4.18.- Justificación del resultado de la conversión en ADRESH: ADRESL.

En ADCON0 también se configuran los pines del Puerta A, para el caso de realizar una conversión vemos el cuadro de la tabla XVIII y configuramos en ADCON0 que pines de la Puerta A queremos como entradas/salidas analógicas o digitales o si alguna de ellas va a servir como voltaje de referencia o si la referencia la escogemos de la misma alimentación del Pic.

En la figura 4.19 vemos el diagrama de bloques del módulo A/D.

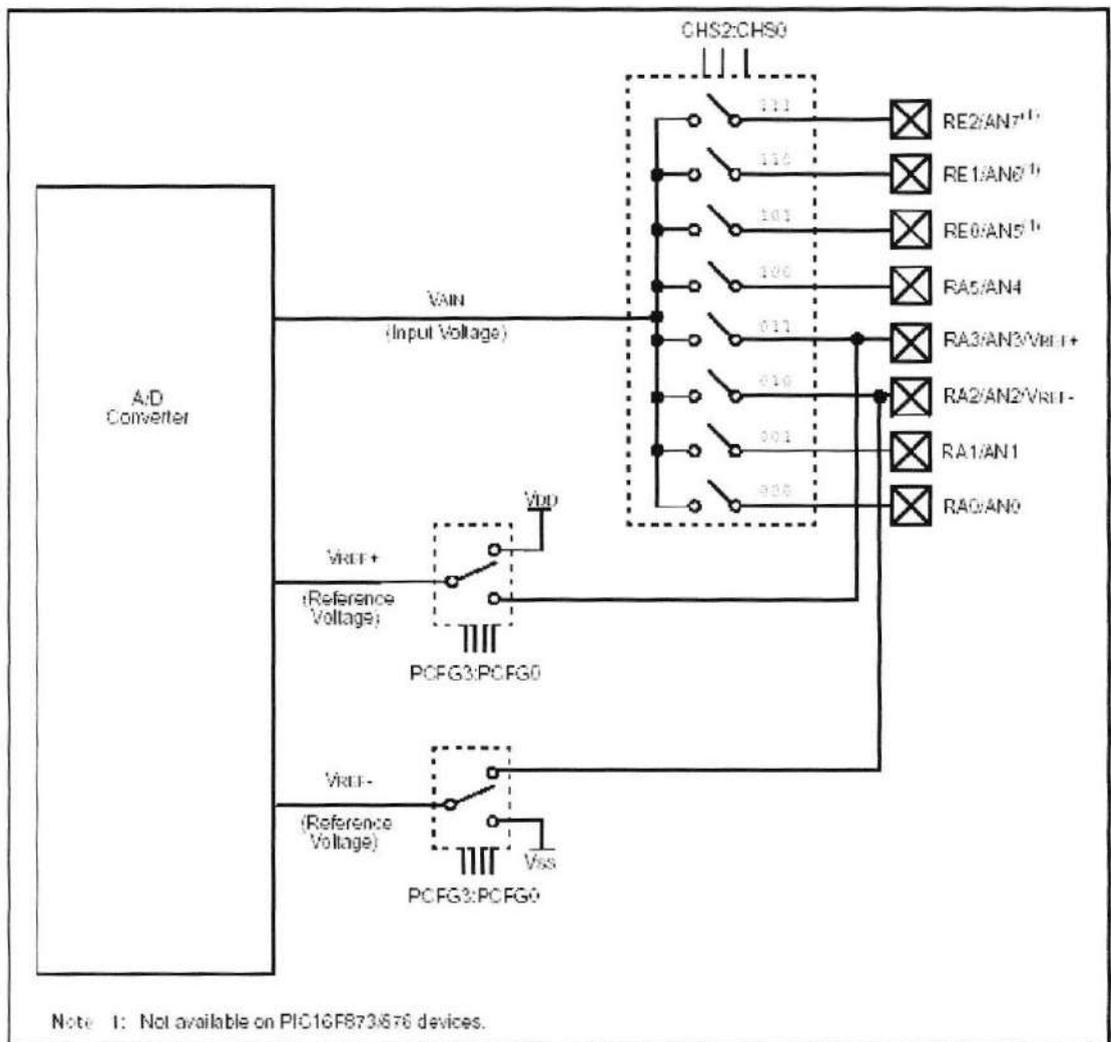


Figura 4.19.- Diagrama de la estructura interna del convertor A/D.

### Requerimientos de adquisición.

Para que el modulo A/D logre la exactitud especifica, el capacitor de captura y mantenimiento (CHOLD) se le debe permitir que cargue hasta el nivel de voltaje en la entrada del canal. El modelo de la entrada analógica se muestra en la figura 4.20:

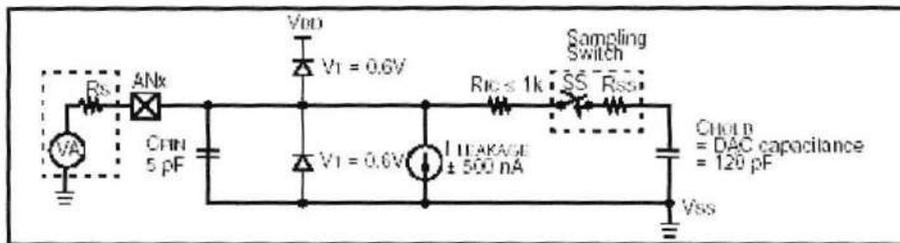


Figura 4.20.- Diagrama de la entrada al convertidor A/D

- CPIN:** Capacitancia de entrada.
- VT:** Voltaje de Umbral
- I LEAKAGE:** Corrientes de fuga debido a varias juntas.
- RIC:** Resistencia de interconexión.
- SS:** Interruptor de captura.
- CHOLD:** Capacitor de captura y mantenimiento (Sample/Hold).
- VA:** Voltaje analógico
- RS:** Impedancia señal analógica.

La impedancia de la señal de entrada y la interna del interruptor de captura afectan directamente el tiempo requerido para cargar el capacitor CHOLD. La impedancia del interruptor de captura (RSS) varía de acuerdo al voltaje de alimentación (Vdd).

La máxima impedancia recomendada para fuentes analógicas es de 10k. Cuando se disminuye la impedancia, el tiempo de adquisición podría ser disminuido. Para calcular el mínimo tiempo de adquisición la ecuación 4.2 puede ser usada:

$$\text{TACQ} = \text{Tiempo de Amplificación estable} + \text{tiempo de carga de CHOLD} + \text{Coeficiente de Temperatura.}$$

Ecuación 4.2.-

Esta ecuación asume que el error usado es  $\frac{1}{2}$  LSb. El error  $\frac{1}{2}$  LSb es el máximo error permitido para que el módulo A/D reúna la resolución especificada. El tiempo de conversión por bit es definido como TAD. La

conversión requiere un mínimo de 12 TAD por 10 bits. El reloj para la conversión se selecciona por software, las cuatro posibles opciones para TAD son:

- 2Tosc
- 8Tosc
- 32Tosc
- Oscilador interno RC (2-6  $\mu$ s)

Para una conversión correcta, el reloj para la conversión (TAD) debe seleccionarse para asegurar un mínimo TAD de 1.6  $\mu$ s. La tabla XIX muestra los tiempos TAD resultantes derivados de la frecuencia de operación del dispositivo y de la fuente del reloj seleccionado.

Tabla XIX

TAD vs. máxima frecuencia de operación del dispositivo



Fuente del reloj	Frecuencia operación dispositivo	
Operation	ADCS1:ADCS0	Max.
2Tosc	00	1.25 MHz
8Tosc	01	5 MHz
32Tosc	10	20 MHz
RC(1, 2, 3)	11	(Note 1)

Los pasos que se deben seguir para realizar una conversión son:

1.- Configurar el modulo A/D:

- Configurar los pines de la Puerta A como entradas analógicas, de referencia (RA2 y RA3), o digitales (ADCON1).
- Seleccionar el canal de entrada (ADCON0)
- Seleccionar la frecuencia del reloj para la conversión (ADCON0)
- Habilitar el modulo A/D (ADCON0)

2.- Configurar las interrupciones del modulo A/D (opcional):

- Limpiar el bit ADIF
- Habilitar el bit ADIE
- Habilitar el bit PEIE
- Habilitar el bit GIE

3.- Esperar el tiempo de adquisición necesario:

4.- Iniciar la conversión:

- Habilitar el bit GO/DONE (ADCON0)

5.- Esperar que la conversión termine, por medio:

- Esperando por el bit GO/DONE esté en cero
- Esperando la interrupción del modulo A/D

6.- Leer el resultado de la conversión en el par de registros (ADRESH: ADRESL)

7.- Para la siguiente conversión vaya al paso 1 o al paso 2 cuantas veces lo requiera. El tiempo de conversión por bit está definido como TAD. Se requiere un mínimo de 2TAD antes de que la siguiente conversión se inicie.

#### **4.1.7 TEMPORIZADORES.**

El Pic16F877A consta de un conjunto de tres temporizadores TMR0, TMR1, TMR2.

*Temporizador TMR0.-*

El TMR0 tiene las siguientes funciones:

1. TMR0 es un contador/temporizador de 8 bits.

2. Se puede Leer y Escribir.
3. Reloj interno y externo.
4. Selección del flanco en el reloj externo.
5. Predivisor de la frecuencia del reloj programable.
6. Generación de interrupción opcional en el desbordamiento.

El funcionamiento del temporizador TMR0 depende de la configuración del registro OPTION. Para que funcione como Temporizador el bit TOCS debe ser limpiado o puesto a uno, el temporizador se incrementa cada ciclo de instrucción sin el predivisor. Si el TMR0 se escribe con algún valor, dicho valor se debe ajustar para tener la temporización deseada. Para el funcionamiento como contador, el bit TOCS debe ser establecido a uno, y se incrementa el contador en cada flanco de subida o bajada en el pin TOCK1, para determinar el flanco si es de subida o bajada se debe configurar el bit TOSE del registro OPTION a uno. El Predivisor es un valor el cual divide la frecuencia del oscilador para obtener el valor deseado de temporización, es asignado al Temporizador TMR0 y al perro guardián (WDT), por medio del bit PSA se establece a que temporizador es asignado y los bits PS2:PS0 determinarán el valor del predivisor.

La figura 4.21 muestra el diagrama de bloques de funcionamiento del TMR0:

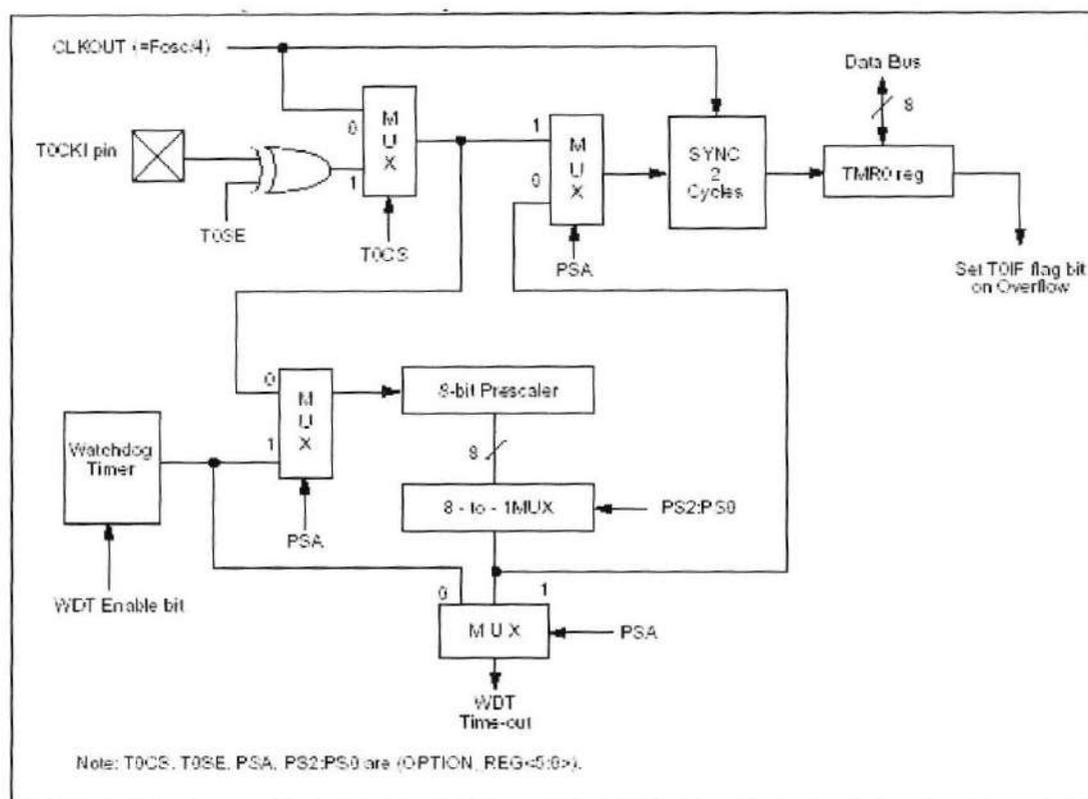


Figura 4.21.- Diagrama de Conexiones del TMR0.

### Temporizador TMR1.-

El TMR1 es más flexible y potente que el timer0 y se caracteriza por:

1. TMR1 es un contador o temporizador de 16 bits.
2. Es leíble y Escribible.
3. Selección de reloj interno o externo.
4. Interrupción opcional por desbordamiento de FFFFh a 0000h.
5. Posible reinicialización desde los módulos CCP.

El TMR1 es un contador/temporizador de 16 bits que consta de dos registros de 8 bits concatenados TMR1H:TMR1L, los cuales guardan el valor del conteo a cada momento cuando funciona. El TMR1 empieza su conteo desde el valor 0000h hasta FFFFh luego de lo cual se activa el señalizador TMR1F y regresa a su valor inicial 0000h, también se puede activar una interrupción para este evento.

El valor contenido TMR1H:TMR1L puede ser leído o escrito y los impulsos de reloj que originan el conteo ascendente pueden provenir del exterior o de la frecuencia de funcionamiento del PIC ( $F_{OSC}/4$ ).

A diferencia del TMR0, funciona de tres formas:

- Temporizador.
- Contador síncrono.
- Contador asíncrono.

El modo de funcionamiento se configura con los bits TMR1CS del registro T1CON y el bit de sincronización T1SYNC. En el modo de temporización aumenta el valor en TMR1H:TMR1L se incrementa con cada ciclo de instrucción ( $F_{OSC}/4$ ). En el modo contador, el incremento se realiza con los flancos ascendentes de un reloj externo, cuya entrada se realiza en los pines RC0 y RC1 de la Puerta C (T1CK1), o por impulsos aplicados en la línea RC0. El registro T1CON sirve para configurar el funcionamiento del TMR1. A continuación se presenta la Tabla XX con todos los bits de control del T1CON:

### T1CON

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	T1SYNC	TMR1CS	TMR1ON	
bit 7								bit 0

Tabla XX

## Bits de control del Registro T1CON

<b>T1CKPS1:T1CKPS0:</b>	<p>Con estos bits se establece el rango del predivisor de frecuencia, un divisor de los impulsos que se aplican a TMR1.</p> <p>11 = 1:8  10 = 1:4  01 = 1:2  00 = 1:1</p>
<b>T1OSCEN:</b>	<p><b>1</b>= Cuando los impulsos externos provienen de un oscilador externo.  <b>0</b>= Cuando los impulsos se reciben en RC0.  En ambos casos se usa cuando funciona como contador de eventos externos por lo que no importa el estado de TRISC ya que RC0 y RC1 se establecen automáticamente como entrada de impulsos.</p>
<b>T1SYNC:</b>	<p>Determina si se puede sincronizar los impulsos del reloj externo con la del reloj interno, según su valor 1 o 0.</p>
<b>TMR1CS:</b>	<p><b>1</b>= Selecciona los impulsos de un reloj externo en RC0 y RC1.  <b>0</b>= Selecciona los impulsos del reloj interno (FOSC/4).</p>
<b>TMR1ON:</b>	<p><b>1</b>= Habilitado el Timer1.  <b>0</b>= Deshabilitado el Timer1.</p>

En la figura 4.22 se muestra el diagrama de bloques del TMR1:

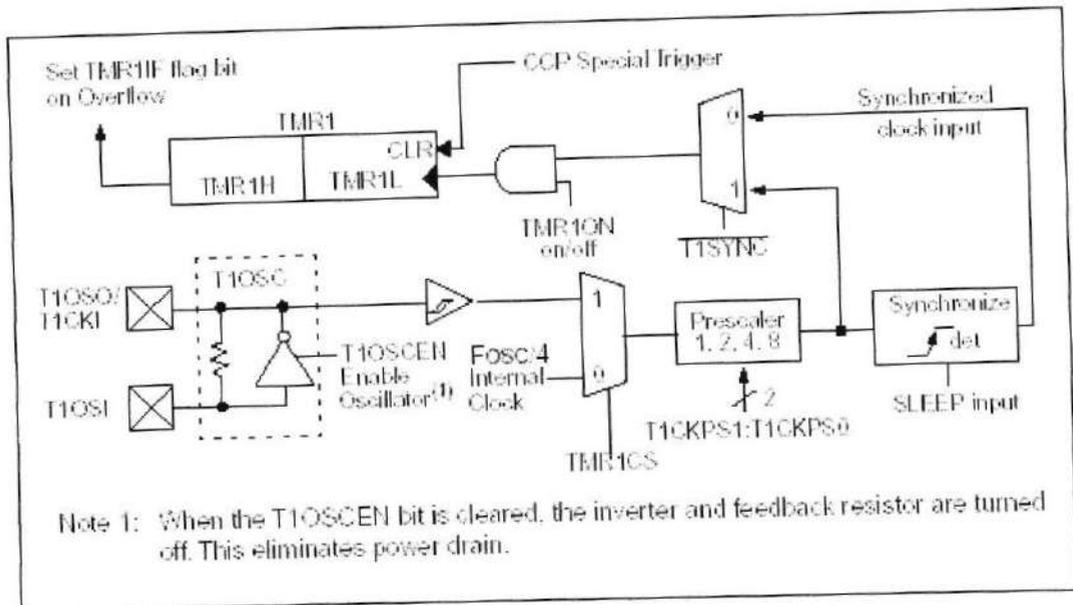


Figura.4.22- Diagrama de Bloques del TMR1.

### Temporizador TMR2.-



CIB-ESPOL

El TMR2 es un temporizador ascendente de 8 bits, se puede leer y escribir, y realiza funciones especiales para el módulo de la Puerta serie sincrónica y los módulos de Comparación y Captura (CCP). La señal de reloj es interna solamente ( $F_{OSC}/4$ ), y antes de ser aplicada pasa por un predivisor y un postdivisor de 1:1 a 1:16 cada uno. Si el Pic se encuentra en modo SLEEP, el Timer 2 deja de funcionar ya que se detiene el oscilador interno. A continuación en la Tabla XXI se describe los bits del registro T2CON que es el registro de control del TMR2:

### T2CON

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
bit 7							bit 0

Tabla XXI

Bits de Control del Registro T2CON.

<b>TOUTPS3:TOUTPS0:</b>	Determina el rango del divisor de frecuencia del postdivisor.
<b>TMR2ON:</b>	<b>1=</b> Habilita el TMR2. <b>0=</b> Deshabilita el TMR2.
<b>T2CKPS1:T2CKPS0</b>	Selecciona el rango de división del predivisor. <ul style="list-style-type: none"> <li>▪ 00= 1:1</li> <li>▪ 01= 1:4</li> <li>▪ 1x= 1:16</li> </ul>

Este temporizador tiene la capacidad de producir una petición de interrupción si se pone a 1 el bit de interrupción TMR2IE del registro PIE1. El TMR2 tiene un registro de Periodo PR2, que cuando el valor de conteo es igual al valor de PR2 se produce un impulso en la salida del comparador EQ que pueden ser divididos

por el postdivisor antes de activar el señalizador TMR2IF. Con la programación adecuada los impulsos de salida del TMR2, antes del postdivisor pueden servir como bases de tiempo para el módulo SSP. La figura 4.23 muestra el diagrama de bloques del TMR2.

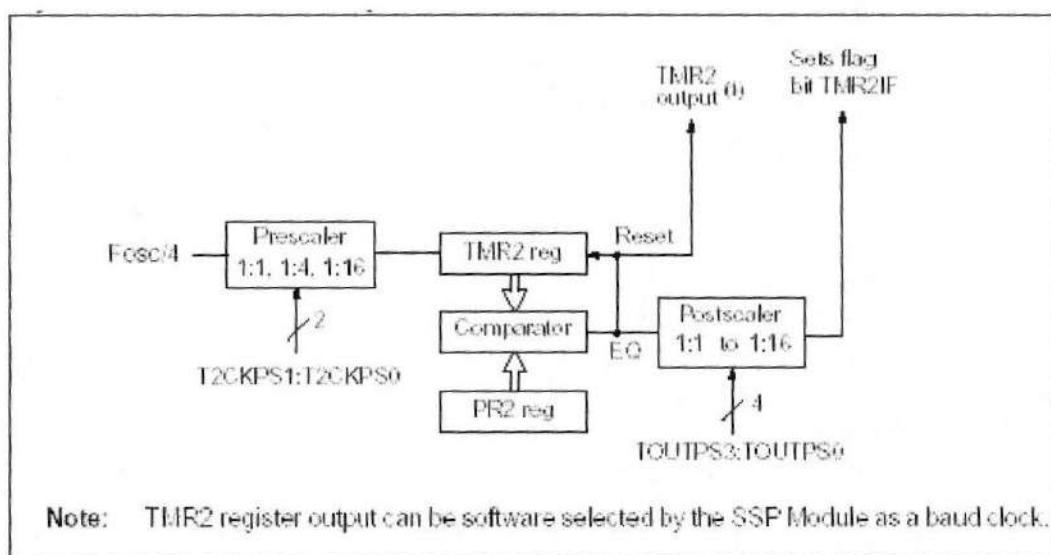


Figura 4.23.- Diagrama de Bloques del TMR2.

#### 4.1.8 COMUNICACION SERIAL I2C.

El protocolo de comunicación I2C fue desarrollado para implementar diversos productos electrónicos que requerían una elevada interconexión de circuitos

integrados. El protocolo I2C utiliza únicamente dos líneas para la transferencia de información entre los elementos que se acoplan al bus (Figura 4.24). Una de dichas líneas se dedica a soportar los datos, es bidireccional y se llama SDA. Los impulsos de reloj los genera el maestro y tienen la función de sincronizar las transferencias con todos los esclavos colgados a las dos líneas.

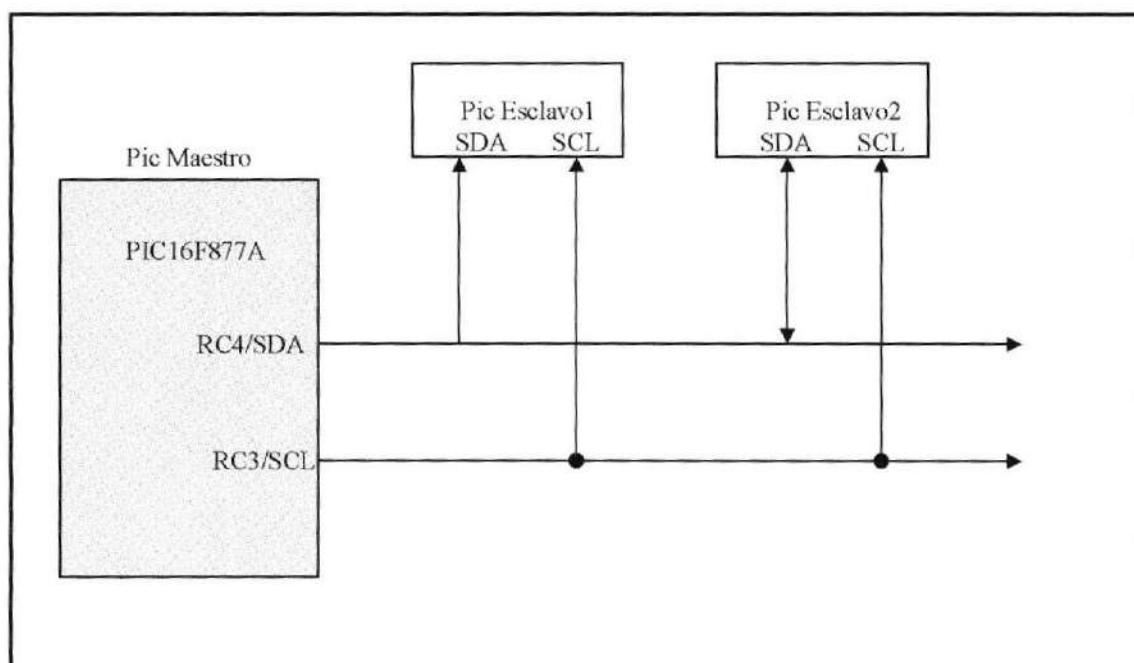


Figura 4.24.- EL Pic maestro se conecta a todos los esclavos en el bus I2C con solo dos líneas SDA de datos y SCL los impulsos del reloj.

### *Concepto del Bus I2C*

En ciertas aplicaciones más completas a parte del microcontrolador necesitan otros periféricos como memorias RAM, EEPROM, ampliación de líneas de E/S, convertidores A/D y D/A, relojes en tiempo real, controladores de display LCD, etc, entonces en sistemas así es bueno usar el bus I2C donde la velocidad de transferencia entre el microcontrolador y los periféricos no es tan esencial y los costos de conexión entre ellos se reduce al mínimo y el hardware es mucho más sencillo.

Los distintos dispositivos conectados al bus serie deben comunicarse entre sí mediante un protocolo que evite el bloqueo de la información y garantice la comunicación entre todos ellos.

El sistema debe ser independiente de los dispositivos conectados al bus, con objeto de poder realizar futuras ampliaciones o modificaciones. Mediante el protocolo adecuado se decide que dispositivo se hace cargo del bus y cuándo.

Dos líneas, SDA (Datos) y SCL (reloj), así transportan la información entre los diferentes dispositivos conectados al bus. Cada dispositivo se identifica por una única dirección y puede transmitir o recibir dependiendo de la operación que se vaya a realizar.

Los dispositivos en un bus I2C pueden clasificarse en Maestro y Esclavo. El maestro es el que inicia la transferencia de datos y genera la señal de reloj. Cualquiera de los dispositivos direccionados por un maestro se considera un esclavo.

El I2C es un bus multi-maestro; puede haber más de un maestro conectado y controlado el bus en la figura 4.25 podemos ver como se conecta un sistema al bus I2C.

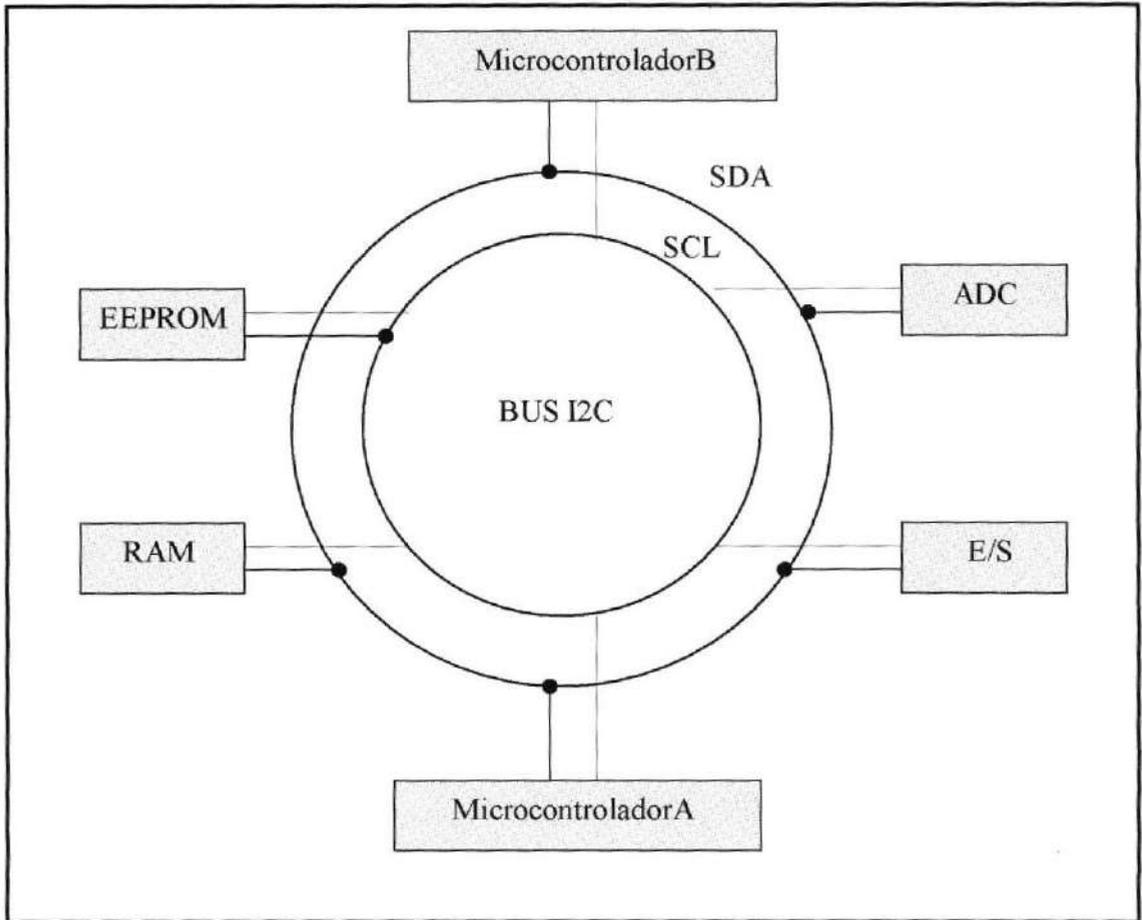


Figura 4.25.- Conexiones al bus.

Según la figura 4.25 podemos tener el caso en que:

1° caso: A envía información al B.

A, emisor, direcciona a B.

A, emisor, envía el dato a B, receptor.

A termina la transferencia.

2° caso: A recibe información desde el B.

A, emisor, direcciona a B.

A, receptor, recibe el dato desde B, transmisor.

A, termina la transferencia.

*El Pic16877A con el modulo I2C.-*

Cada transferencia comienza con la condición de inicio START y termina con la condición de parada STOP. Ambas condiciones las genera el maestro, y la primera consiste en un flanco descendente en SDA mientras SCL tiene un nivel alto. La condición de parada es un flanco ascendente mientras SCL tiene un nivel alto.

Los datos que se colocan sobre la línea SDA son bytes de 8 bits, que comienzan con el MSB y terminan con el bit de menos peso, al cual le sigue el noveno bit, la condición de reconocimiento ACK, en este caso el transmisor pone SDA = 1, mientras que el receptor pone SDA = 0 en dicho impulso de reloj,

prevaleciendo sobre la línea el nivel bajo, este bit se usa para saber que se ha recibido el dato.

El primer byte que envía el maestro tras el START es la dirección del esclavo con el que se va a realizar la comunicación. EL código cero se usa para realizar una llamada general sobre todos los esclavos. La dirección consta de 7 bits, por lo que el octavo bit es para indicar el tipo de operación que se va a realizar  $R/\overline{W}$  (Lectura/Escritura).

El módulo I2C contiene todas las funciones en modo maestro y esclavo y suministra interrupciones en los bits de START y STOP para determinar que el bus está libre.



CIR-ESPOL

Este módulo implementa direccionamientos de 7 bits y 10 bits y trabaja en modos de frecuencia de 100 KHZ y 400 KHZ.

Las líneas SDA y SCL son automáticamente configuradas cuando el módulo I2C es habilitado pero previamente los pines de la Puerta C (RC3/SCL) y (RC4/SDA) deben estar configuradas como entradas en el registro TRISC y para hacerlo hay que poner el bit SSPEN = 1 (SSPCON<5>). El módulo I2C utiliza 6 registros, ellos son:

- Registro de Control (SSPCON).
- Registro de Control 2 (SSPCON2).
- Registro de Estado (SSPSTAT).
- Buffer de Recepción/Transmisión serial (SSPBUF).
- Registro de desplazamiento no accesible directamente (SSPSR).
- Registro de dirección (SSPADD).

Los 4 bits de menos peso del registro SSPCON sirven para seleccionar el modo de trabajo entre estos tres posibles:

- Modo I2C maestro,  $\text{reloj} = (\text{SSPADD} + 1) \cdot \frac{\text{osc}}{4}$ .
- Modo I2C esclavo con 7 bits para la dirección.
- Modo I2C esclavo con 10 bits para la dirección.

El bit CKE (SSPSTAT<6:7>) establece los niveles de los pines SDA y SCL en el modo maestro o esclavo. Cuando CKE = 0 los niveles conformaran las especificaciones del bus I2C.

El registro SSPSTAT nos da el estado de la transferencia de datos, en esta información incluye detección de un bit START o STOP, especifica si el byte recibido fue un dato o dirección, si el siguiente byte es la terminación de una dirección de 10 bits, y si este será una transferencia de escritura o lectura de datos.

El registro SSPBUF es en el que se va a escribir o leer el dato resultado de la transferencia. El registro SSPSR desplaza el dato dentro o fuera del dispositivo. En operaciones de recepción, el SSPSR realiza la conversión serie/paralelo y luego lo pasa a SSPBUF, esto permite la recepción del siguiente byte empiece antes de la lectura del último byte del dato recibido. Cuando el byte completo es recibido en SSPSR, luego es transferido a SSPBUF y se activa la banderita SSPIF. Si otro byte es recibido antes de que SSPBUF sea leído, ocurrirá un

desbordamiento y el bit SSPOV (SSPCON<6>) se activará y el byte en SSPSR se perderá.

El registro SSPADD mantiene la dirección del esclavo. En el modo de direccionamiento de 10 bits, el usuario necesita escribir la parte alta de la dirección con el código (1111 0 A9 A8 0), donde A9:A8 son los bits de más peso. Seguido la el byte bajo debe ser cargado.

#### *Modo Esclavo*

En el modo esclavo, las líneas SDA y SCL deben ser configuradas como entradas. Cuando una dirección recibida coincide con la del esclavo, el hardware automáticamente generará una condición de reconocimiento (ACK), y luego cargará en SSPBUF el valor recibido en SSPSR. En la figura 4.26 se muestra el módulo I2C, modo esclavo:

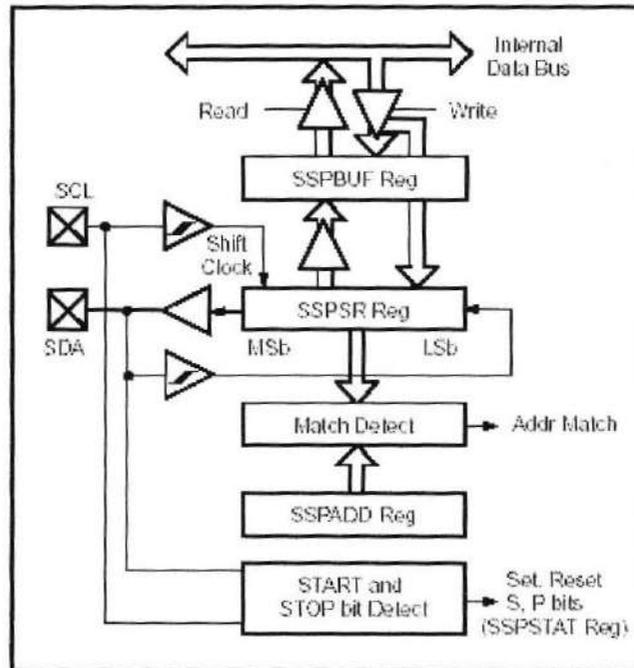


Figura 4.26.- Diagrama de bloques del módulo I2C modo esclavo.

Hay ciertas condiciones que causarán que el módulo I2C no proporcione un ACK, esto ocurrirá si:

- El bit BF (SSPSTAT<0>) se activó antes que la transferencia fuera recibida.
- El bit SSPOV (SSPCON<6>) fue activado antes que la transferencia fuera recibida.

Si se activa el bit BF, el valor en el registro SSPSR no se carga en SSPBUF, pero el bit SSPIF y SSPOV son activados también.

En la tabla se XXII puede observar que pasa con el dato dando el estado de los bits BF y SSPOV

Tabla XXII

Acciones del dispositivo cuando se recibe un byte según el estado de BF y SSPOV.

Estado de los bits cuando un dato es recibido.		SSPSR → SSPBUF	GENERACION ACK	ACTIVACION SSPIF
BF	SSPOV			
0	0	SI	SI	SI
1	0	NO	NO	SI
1	1	NO	NO	SI
0	1	SI	NO	SI

El bit BF se pone a cero leyendo el registro SSPBUF, mientras que SSPOV a través de software. La línea SCL debe tener un mínimo de tiempo en estado alto y en bajo para una correcta operación, según las especificaciones del módulo I2C.

### *Direccionamiento*

Una vez el modulo I2C ha sido habilitado, esperará que ocurra una condición de START, seguido a esto, los 8 bits se desplazarán dentro del registro SSPSR. Todos los bits entrantes son capturados en la subida de la línea SCL (reloj). El valor del registro SSPSR<7:1> es comparado con el valor en el registro SSPADD. La dirección es comparada a la bajada del octavo pulso de reloj (SCL). Si la dirección coincide, y BF y SSPOV están en cero, los siguientes eventos ocurren:

- a) EL valor en SSPSR es cargado en SSPBUF en la bajada del octavo pulso de reloj en SCL.
- b) El bit BF, se pone a uno en la bajada del octavo pulso de reloj en SCL.
- c) Una condición de reconocimiento (ACK) es generada.
- d) La bandera SSPIF (PIR1<3>), se activa en la bajada del noveno pulso de reloj en SCL.

En el modo de direccionamiento de 10-bits, dos bytes de dirección necesitan ser recibidos por el esclavo. Los 5 MSB de la primer byte de dirección especifican si es una dirección de 10-bits. El bit  $R/\overline{W}$  (SSPSTAT<2>) debe especificar una escritura así el dispositivo esclavo recibirá el segundo byte de dirección.

Para una dirección de 10-bits, el primer byte sería igual '1111 0 A9 A8 0', donde A9 y A8 son los dos bits más significativos (MSB) de la dirección. La secuencia de eventos para una dirección de 10-bits es como sigue, con los pasos 7-9 son para modo esclavo transmisor:

1. Recibir el byte alto de la dirección (los bits SSPIF, BF y UA (SSPSTAT<1>) se activan).
2. Actualizar el registro SPADD con el byte bajo de la dirección (poner a cero el bit UA y liberar la línea SCL).
3. Leer el registro SSPBUF (el bit BF se encera) y poner a cero la bandera SSPIF.
4. Recibir el byte bajo de la dirección (los bits SSPIF, BF y UA son activadas).
5. Actualizar el registro SSPADD con el byte alto de la dirección. Esto pondrá a cero el bit UA y luego liberar la línea SCL.

6. Leer el registro SPBUF y poner a cero la bandera SSPIF.
7. Recibir la condición de repetición de inicio.
8. Recibir el byte alto de la dirección (los bits SSPIF y BF son activados).
9. Leer el registro SSPBUF y poner a cero la bandera SSPIF.

En el módulo I2C en modo esclavo puede trabajar en recepción y transmisión.

#### *Esclavo en Recepción.-*

Cuando el bit  $R/\overline{W}$  del byte de dirección está en cero y coincide con la dirección del dispositivo esclavo, el bit  $R/\overline{W}$  del registro SSPSTAT se pone a cero, luego la dirección recibida es cargada en el registro SSPBUF.

Luego de que la dirección coincide y de cargarse en SSPBUF se genera una condición de reconocimiento ( $\overline{ACK}$ ) de que se recibió la dirección, sólo si las condiciones ya mencionadas se cumplen, que BF o el bit SSPOV no estén activados, esto suele ser un error de software que hay que tener en cuenta.



Luego de generar la condición de reconocimiento, por cada byte recibido se activa la bandera bit SSPIF (PIR1<3>), la cual debe ser puesta a cero vía software. Para saber el estado de la transmisión del byte recibido podemos utilizar el registro SSPSTAT.

#### *Esclavo en transmisión.-*

Cuando el bit  $R/\overline{W}$  de la dirección entrante es uno y la dirección del esclavo coincide con la que envía el maestro, el bit  $R/\overline{W}$  de SSPSTAT se pone a uno.

La dirección recibida es cargada en el registro SSPBUF. La condición de reconocimiento  $\overline{ACK}$  será enviada en noveno bit y la línea SCL se mantendrá en un bajo. El dato a transmitir debe ser cargado en SSPBUF, el cual a su vez lo carga en el registro SSPSR. Luego la línea SCL debe ser habilitada, poniendo a uno el bit CKP (SPCON<4>), mientras no lo se lo haga podemos tener esperando al maestro que envíe otro pulso de reloj.

Por cada byte enviado la bandera SSPIF se activa, la cual debe ser encendida por software y el registro SSPSTAT para determinar el estado de la transmisión.

Como un esclavo transmisor, el  $\overline{ACK}$  generado por el maestro receptor es capturado en el noveno impulso de reloj de la línea SCL. Si la línea SDA está en alto ( $\overline{ACK}$  no generado), entonces la transferencia está completa. Cuando  $\overline{ACK}$  no generado es capturado por el esclavo, la lógica del esclavo provocará un RESET al dispositivo y empieza a esperar otra condición de START. Si la línea estuvo en bajo ( $\overline{ACK}$  generado). El dato transmitido debe ser cargado en SSPBUF, el cual además lo carga en el registro SSPSR. Luego la línea SCL debe ser habilitada poniendo a uno el bit CKP.

### *Modo Maestro*

Todos los dispositivos maestros (figura 4.27), generan pulsos de reloj y las condiciones de START y STOP. Una transferencia es terminada con una condición de STOP.

En el modo maestro transmisor, los datos seriales salen a través de SDA, mientras que los pulsos de reloj por SCL. El primer byte transmitido contiene la dirección de esclavo del dispositivo receptor de 7 bits y el bit Lectura/Escritura ( $R/\overline{W}$ ). En este caso el bit  $R/\overline{W}$  será un cero. Los datos son transmitidos en paquetes de 8 bits. Después de que cada byte es transmitido, una condición de reconocimiento ( $\overline{ACK}$ ) es recibido. Las condiciones de START y STOP sirven para indicar el comienzo y final de una transferencia serial.

En el modo maestro receptor, el primer byte contiene la dirección del esclavo, 7 bits de datos y el bit  $R/\overline{W}$  que en este caso es un uno. Después de que cada paquete de 8 bits es recibido, una condición de reconocimiento ( $\overline{ACK}$ ) es transmitida.

El generador de BAUD usado por el módulo I2C es usado para la frecuencia del reloj en SCL que puede ser de 100 Khz., 400 Khz. o 1 MHz. El generador de BAUD recarga su valor que está contenida en los 7 bits de menos peso del

registro SSPADD y automáticamente empezará a contar en una operación dada.

Una vez la operación dada es completada, el último dato es seguido por un  $\overline{ACK}$ , el reloj interno automáticamente parará de contar y la línea SCL mantendrá su último estado.

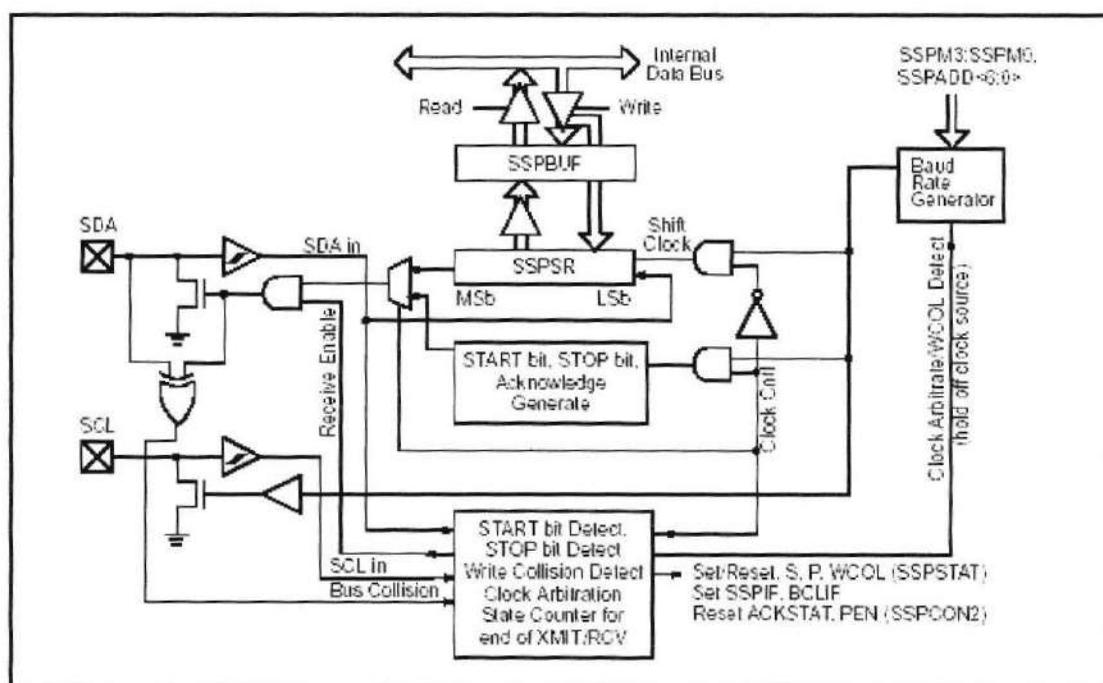


Figura 4.27.- Estructura interna del módulo I2C modo maestro.

Una típica secuencia de transmisión es:

- a) El usuario debe generar la condición de START poniendo a uno el bit SEN en SPCON2.
- b) La bandera SSPIF se activa. El módulo esperará el tiempo requerido antes de que otra operación tome lugar.
- c) El usuario carga SSPBUF con la dirección a transmitir.
- d) Los 8 bits de la dirección salen por la línea SDA.
- e) El módulo recoge el bit ACK de el dispositivo esclavo y escribe su valor en el registro SSPCON2 (SSPCON2<6>).
- f) El módulo genera una interrupción al final del noveno pulso del reloj se activa la bandera SSPIF.
- g) El usuario carga en el registro SPBUF con 8 bits de datos.
- h) Por la línea SDA se desplaza el dato hasta que los 8 bits sean transmitidos...
- i) El módulo recoge el bit ACK del dispositivo esclavo y escribe su valor en SSPCON2.
- j) El módulo genera una interrupción al final del noveno pulso del reloj cuando se activa la bandera SSPIF.
- k) El usuario genera una condición de STOP poniendo a uno el bit PEN en SSPCON2.
- l) Una interrupción es generada una vez la condición de STOP es completada.

### *Modo Maestro en transmisión*

La transmisión de un byte de datos debe ir precedida de la transmisión de la dirección del esclavo, la cual debe cargarse en el registro SSPBUF, en cuyo momento se pone el bit BF = 1 y se pone en marcha el reloj y el inicio de la transmisión. Cada bit que sale por la línea SDA lo hace en el flanco descendente del impulso de reloj en SCL. Tras sacar los 8 bits se borra el bit BF y el esclavo debe generar la condición de reconocimiento ( $\overline{ACK}$ ) en el noveno impulso de reloj. El bit  $\overline{ACK}$  se coloca en el bit ACKDT en el flanco descendente de SCL. Tras el noveno impulso SSPIF = 1 y se detiene el generador de frecuencia hasta que no se cargue otro dato en SSPBUF. El señalizador WCOL vale 1 cuando el usuario escribe sobre el SSPBUF estando una transmisión del dato previo en progreso. Este bit hay que pasarlo a 0 por software. El bit ACKSTAT pasa a cero cuando el esclavo ha enviado el bit de reconocimiento.

### *Modo Maestro en recepción*

Para poner en marcha el modo maestro en recepción, hay que poner el bit RCEN = 1 estando en reposo el módulo I2C. El generador de frecuencia (BAUD) empieza a funcionar y SCL cambia de estado en cada periodo controlado por el decremento del contador BRG. Entonces el dato empieza a desplazarse sobre el registro SSPSR. Al final del octavo pulso de reloj se ponen a BF = SSPIF = 1 y se transfiere el contenido de SSPSR al SSPBUF, con el RCEN = 0. El módulo queda en estado de reposo a la espera de un nuevo comando. Cuando el buffer es leído BF se pone a cero automáticamente. El usuario puede enviar un bit de reconocimiento al finalizar la recepción, escribiendo ACKEN = 1 (SSPCON2<4>).

Cuando un dato o una dirección se carga en SSPBUF desde SSPSR, el bit BF = 1 y este bit pasa a cero cuando se lee SSPBUF. La bandera de desbordamiento SSPOV es 1 cuando se recibe un byte en SSPSR y aún no se ha leído el dato anterior.

Si el usuario escribe en SSPBUF mientras hay una recepción en progreso, el bit WCOL se pone a uno y el contenido del buffer no cambia.

#### **4.1.9 COMUNICACION USART.**

El USART, llamado SCI (Serial Communications Interface), es uno de los dos módulos seriales. El USART puede ser configurado como en sistema asincrónico *full duplex* que se puede comunicar con periféricos como monitores CTR y computadores personales, o puede ser configurado como un sistema sincrónico *half duplex* que se puede comunicar con periféricos como convertidores A/D o D/A, memorias seriales EEPROMS etc.

El USART puede ser configurado como los siguientes modos:

- Asincronico (full duplex)
- Síncrono - Maestro (half duplex)
- Síncrono - Esclavo (half duplex)

El bit SPEN (RCSTA<7>) y los bits TRISC<7:6> tienen que ser puestos a uno para configurar los pines RC6/TX/CK y RC7/RX/DT para usar el módulo USART.

Los registros para el control del USART se ven en la Tabla XXIII y XXIV son:

### REGISTRO TXSTA

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-1	R/W-0
CSCRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D
bit 7							bit 0

Tabla XXIII

Bits del control del registro TXSTA.

<b>CSCRC:</b>	Bit de selección del reloj. Modo asíncrono: no influye. Modo síncrono: <b>1</b> = Modo maestro (reloj generado internamente desde BRG). <b>0</b> = Modo esclavo (reloj generado por una fuente externa).
<b>TX9:</b>	Habilita el bit 9 de transmisión. <b>1</b> = Selecciona transmisión de 9 bits. <b>0</b> = Selecciona transmisión de 8 bits.
<b>TXEN:</b>	Activa la transmisión.
<b>SYNC:</b>	Bit de selección del modo USART.

	<b>1=</b> Modo síncrono. <b>0=</b> Modo asíncrono
<b>BRGH:</b>	Bit de selección de la velocidad de baudios. Modo asíncrono: <b>1=</b> Alta velocidad. <b>0=</b> Baja velocidad. Modo síncrono: no utilizado.
<b>TRMT:</b>	Bit de estado del registro de desplazamiento de transmisión. <b>1=</b> TSR vacío. <b>0=</b> TSR no vacío.
<b>TX9D:</b>	Bit 9 del dato a transmitir

### REGISTRO RCSTA

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	OEERR	RX9D
bit 7							bit 0



Tabla XXIV

Bits del control del registro RCSTA.

<b>SPEN:</b>	Habilitación del puerto serie... <b>1</b> = Habilitado (se configuran los pines RC7 y RC6). <b>0</b> = Deshabilitado.
<b>RX9:</b>	Habilita el bit 9 de Recepción... <b>1</b> = Selecciona recepción de 9 bits. <b>0</b> = Selecciona recepción de 8 bits.
<b>SREN:</b>	Activa la recepción sencilla.
<b>CREN:</b>	Configura la recepción continua. Modo asíncrono: <b>1</b> = Habilita modo de recepción continua. <b>0</b> = Deshabilita recepción continua. Modo síncrono: <b>1</b> = Habilita recepción continua hasta que el bit CREN es borrado. <b>0</b> = Deshabilita recepción continua.
<b>ADDEN:</b>	Detección de dirección. Modo asíncrono con 9 bits (RX9 = 1): <b>1</b> = activa la detección de dirección, activa la interrupción y descarga el buffer de recepción al activarse RSR<8>. <b>0</b> = Desactiva la detección de dirección, todos los bits son recibidos y el bit 9 puede ser utilizado como bit de paridad.
<b>FERR:</b>	Bit de error de trama. <b>1</b> = Error de trama (puede ser actualizado leyendo el registro RCREG y recibir el siguiente dato válido). <b>0</b> = No hay error de trama.
<b>OERR:</b>	Bit de error de sobrepasamiento
<b>RX9D:</b>	.Bit 9 del dato recibido (puede ser el bit de paridad).

En la figura 4.28 se muestra un esquema del USART en modo síncrono y asíncrono. En el primero, la transferencia de información se realiza sobre dos líneas TX (Transmisión) y Rx (Recepción), saliendo y entrando los bits por dichas líneas al ritmo de una frecuencia controlada internamente por el USART.

En este proyecto se utilizó la norma más común de comunicación RS-232, donde cada palabra de información o dato se envía independiente de los demás. Suele constar de 8 o 9 bits y van precedidos por un bit de START (inicio) y detrás de ellos se coloca un bit de STOP (parada). Los bits se transfieren a una frecuencia fija y normalizada.

El módulo tiene cuatro bloques, en modo asíncrono son:

1. Circuito de Muestreo.
2. Generador de Baudios.
3. Transmisor asíncrono.
4. Receptor Asíncrono.



### *1. Circuito de Muestreo.*

El circuito de muestreo actúa sobre el pin RC7/RX/DT, que es por donde recibe el bit de información o de control y se encarga de muestrear tres veces su valor, para decidir por mayoría.

### *2. Generador de Baudios*

En el protocolo asíncrono RS-232, la frecuencia en baudios (bits por segundo) a la que se realiza la transferencia se debe efectuar a un valor normalizado; 30, 600, 1200, 2400, 4800, 9600, 19200, 38400, etc. Para generar esta frecuencia, El USART dispone de un Generador de frecuencia en BAUDIOS, BRG, cuyo valor es controlado por el contenido grabado en el registro SPBRG. Además del valor X cargado en el registro SPBRG, la frecuencia en baudios del bit BRGH del registro TXSTA<2>. En el caso de que BRGH sea cero se trabaja en baja velocidad y si BRGH = 1 se trabaja a alta velocidad. Según este bit se obtendrá el valor de una constante K necesaria en la determinación de la frecuencia de funcionamiento según la ecuación 4.3.

$$\text{Frecuencia en Baudios} = \text{Fosc} / (K * (X+1))$$

Ecuación 4.3.-

Donde X es el valor en el registro SPBRG.

Si BRGH = 0, baja velocidad y K = 64.

Si BRGH = 1, alta velocidad y K = 16.

Donde:

$$X = \text{Fosc} / (\text{Frecuencia} / K - 1)$$

Ecuación 4.4.-

La Tabla XXV muestra los valores de X en SPBRG, para determinar ciertas velocidades:

Tabla XXV

Baudios generados por el modo asíncrono con 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	19.631	1.72	31
28.8	29.070	0.94	42	29.412	2.13	33	28.409	1.36	21
33.6	33.784	0.65	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	1.2	0	191
2.4	2.404	0.17	103	2.4	0	95
9.6	9.615	0.16	25	9.6	0	23
19.2	19.231	0.16	12	19.2	0	11
28.8	27.798	3.65	8	28.8	0	7
33.6	35.714	6.29	6	32.9	2.04	6
57.6	62.500	8.51	3	57.6	0	3
HIGH	0.977	-	255	0.9	-	255
LOW	250.000	-	0	230.4	-	0



CIB-ESPOL

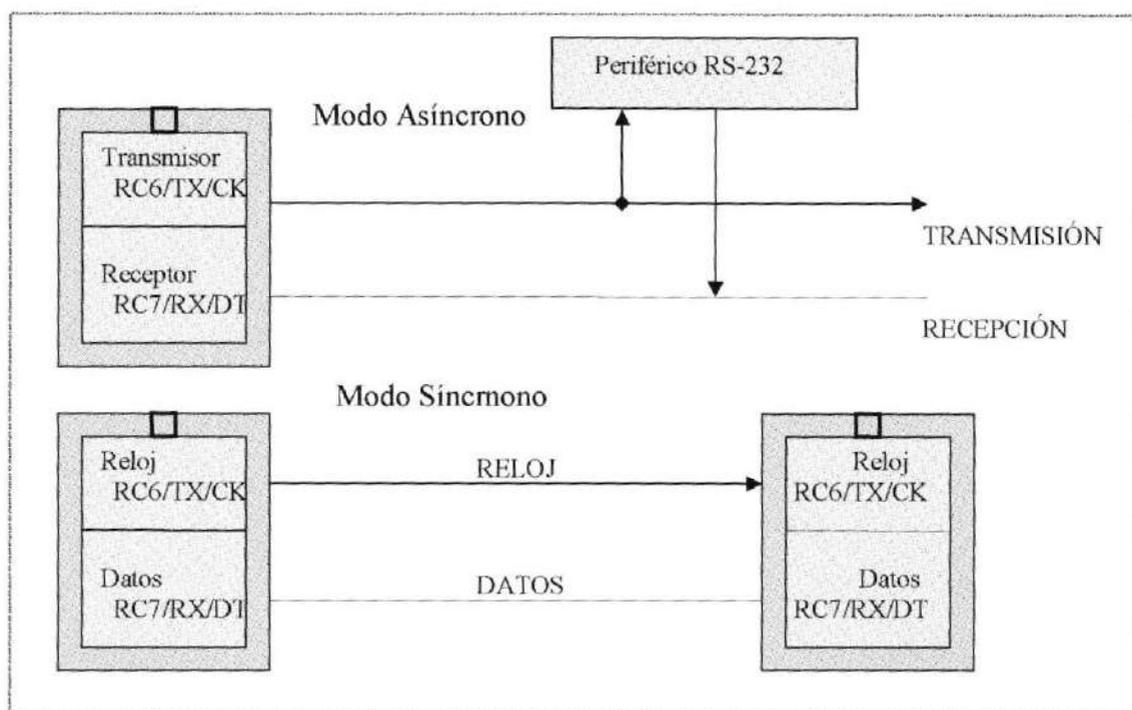


Figura 4.28.- Esquema de la comunicación USART.

### *Transmisor Asíncrono*

El dato que se desea transmitir por el USART transmisor se deposita en el registro TXREG y a continuación se traspasa al registro de desplazamiento TSR, que va sacando los bits secuencialmente y a la frecuencia establecida. Además, antes de los bits del dato de información incluye un bit de inicio y después de sacar todos los bits añade un bit de PARADA. El USART receptor recibe, uno a uno, los bits, elimina los dos de control y los de información una

vez que han llenado el registro de desplazamiento RSR los traslada automáticamente al registro RCREG, donde quedan disponibles para su posterior procesamiento.

El corazón de este módulo está constituido por el registro desplazamiento RSR, que obtiene el dato desde el registro TXREG y luego lo va desplazando y sacando bit a bit, en serie, por el pin RC6/TX/CK. El primer bit que sale es el de menos peso. El dato a transferir se carga por software en TXREG y se transfiere al TSR en cuanto se haya transmitido el bit de parada del dato anterior. La transferencia entre los registros se realiza en un ciclo y entonces la bandera TXIF se pone a uno, para advertir que el registro de transmisión se ha vaciado. También en este momento puede producirse una interrupción si se ha habilitado el bit TXIE = 1 (PIE1<4>).



CIB-ESPOL

Cuando se escribe otro dato sobre TXREG, la bandera TXIF se pone a cero. EL bit TRMT sirve para indicar el estado del registro TSR y vale 1 cuando está vacío.

La figura 4.29 muestra el diagrama de conexiones del módulo USART:

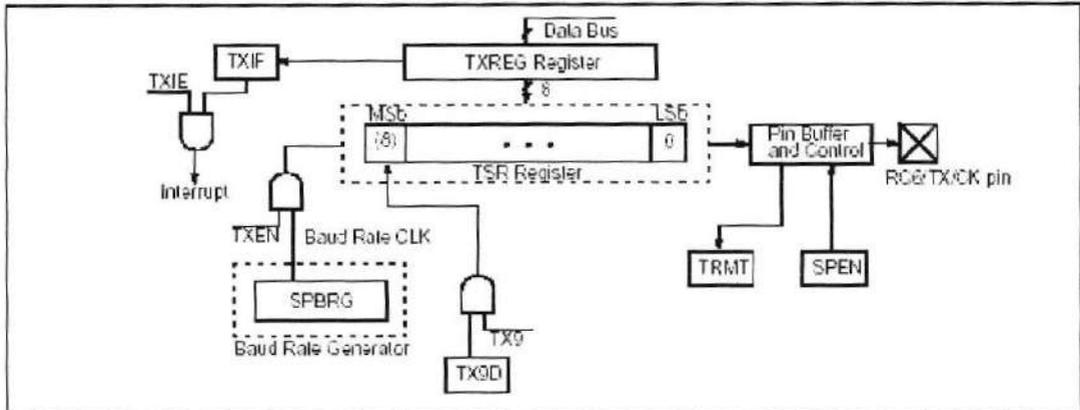


Figura 4.29.- Diagrama del módulo USART como transmisor.

La secuencia de pasos a seguir para una transmisión la siguiente:

1. Hay que configurar las líneas RC6 como salida y RC7 como entrada.
2. Poner SYNC = 0 y SPEN = 1 para activar el USART en modo asíncrono.
3. Si se desea trabajar con interrupción, poner TXIE = 1, además de habilitar las interrupciones en general.
4. Si el dato consta de 9 bits, en lugar de los 8 típicos, poner el bit TX9 = 1. El noveno bit se coloca en TXD9 (TXSTA).

5. Se carga el valor X adecuado en el registro SPBRG, para producir la frecuencia de trabajo deseada. Hay que controlar el bit BRGH (alta y baja velocidad).
6. Activar la transmisión con TXEN = 1. El bit TXIF tendrá valor 1, ya que TXREG se encuentra vacío.
7. Cargar en TXREG el dato a transmitir. Comienza la transmisión.

#### *Receptor Asíncrono*

Los datos se reciben en serie, bit a bit, por la patita RC7/RX/DT y se van introduciendo secuencialmente en el registro desplazamiento RSR, como se ve en la figura 4.30, que funciona a una frecuencia 16 veces más rápida que la de trabajo. Cuando el dato consta de 9 bits hay que programar el bit RX9 = 1 y el noveno bit de información se colocará en el bit RX9D del registro RCSTA. Cuando CREN = 1 en el registro RCSTA<4>, se habilita la recepción.

Cuando un procesador maestro intenta enviar información a uno de los esclavos, primero envía un byte de dirección que identifica al destinatario. El

byte de dirección se identifica porque el bit RX9D que llega vale 1. Si el bit ADDEN = 1 en el esclavo se ignoran todos los bytes de datos. Pero si el noveno bit que se recibe es uno, quiere decir que se trata de una dirección y el esclavo provocará una interrupción, y se transferirá el contenido del registro RSR al buffer de recepción.

Tras la interrupción, el esclavo deberá examinar la dirección y si coincide con la suya poner ADDEN = 0 para poder recibir datos del maestro.

Si ADDEN = 1 como los datos son ignorados, el bit de PARADA no se carga en RSR, por lo que este hecho no produce interrupción.

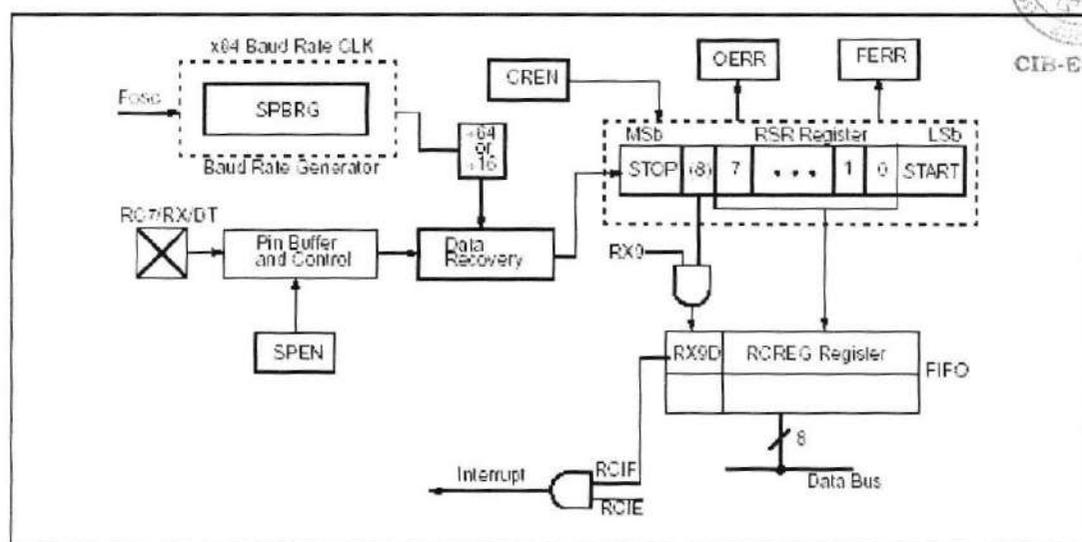


Figura 4.30.- Diagrama de conexiones del USART como Receptor

Los pasos a seguir para la recepción son:

1. Se carga con el valor X al registro SPBRG para trabajar con la frecuencia deseada, controlando además el valor de BRGH.
2. Se habilita el USART en modo asíncrono con SPEN = 1 y SYNC = 0.
3. Si se desea que se genere una interrupción con la llegada del bit de PARADA, se pone RCIE = 1, además de habilitar las interrupciones en general.
4. Poner RX9 = 1 para permitir la recepción del bit 9.
5. Para detectar la dirección, poner ADDEN = 1.
6. Se habilita la recepción poniendo CREN = 1.
7. Al completarse la recepción RCIF se pondrá a uno y se produce una interrupción si se había permitido.
8. Se lee el registro RCSTA y se averigua si se ha producido algún error.
9. Leer los 8 bits del registro RCREG para determinar si el dispositivo ha sido el diseccionado.
10. Si se ha producido algún error, poner CREN = 0.

11. Si ha sido diseccionado el dispositivo, poner ADDEN = 0 para permitir la recepción de la información.

#### **4.1.10 CARACTERISTICAS ESPECIALES**

##### *Características del Reset*

La lógica del reset es usado para reiniciar el dispositivo en el vector 0000h. La fuente del reset puede determinarse usando los bits del registro STATUS. La lógica del reset es diseñada con características que reducen los costos del sistema e incrementa la confiabilidad del mismo. Las diferentes causas del reset se enumeran a continuación:

- a) Power-on-Reset (POR), reset por conexión a la alimentación, cuando Vdd crece de 1.2 a 1.7...
- b) Activación del pin MCLR# durante una normal operación con un nivel bajo.
- c) Activación del pin MCLR# durante el SLEEP (modo de bajo consumo).

- d) Por desbordamiento del perro guardián (WDT) en operación normal.
- e) Por desbordamiento del perro guardián (WDT) en SLEEP.
- f) Brown-on-Reset (BOR), reset por caída en la alimentación Vdd baja de 3,8 y 4,2V.

Los bits TO# y PD# del registro STATUS, toman distintos valores de acuerdo a la naturaleza del reset, también los bits de PCON, POR y BOR, se afectan con un reset, por lo tanto en el Firmware se puede hallar la causa de reset.



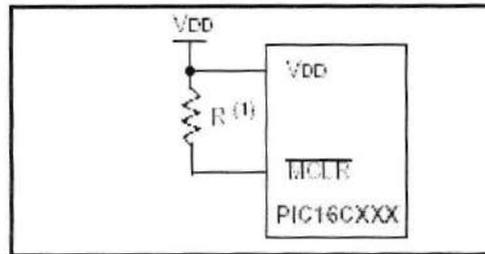


Figura 4.32.- Conexión Externa para generar un Reset.

### Temporizador PWR

El temporizador PWRT (Power-up-Timer) proporciona un retardo al reset fijo de 72 ms es decir que se la da ese tiempo a Vdd para que se estabilice. El PWRT opera con un oscilador RC interno. El dispositivo se mantiene en reset mientras el PWRT está activado.

Para habilitar el temporizador PWRT, el bit PWRTE# debe ser puesto a cero, también debería siempre estar habilitado cuando se habilita el BOR (Brown-on-Reset).

### Temporizador OST

El temporizador OST (Oscillator Start-up Timer) proporciona un retardo de 1024 ciclos de la frecuencia aplicada a la patita OSC, después de que el retardo del PWRT haya terminado, con el objetivo de asegurar que el cristal de cuarzo del oscilador principal esté estabilizado y en marcha.

La figura 4.33 muestra el gráfico, donde MCLR# está conectado a Vdd, cuando Vdd sube el POR# interno se activa pero el reset no se provoca todavía, espera que el temporizador PWRT termine su temporización, luego del cual, el temporizador OST empieza su conteo para asegurar la estabilización del cristal de cuarzo, luego del cual se provoca el reset.

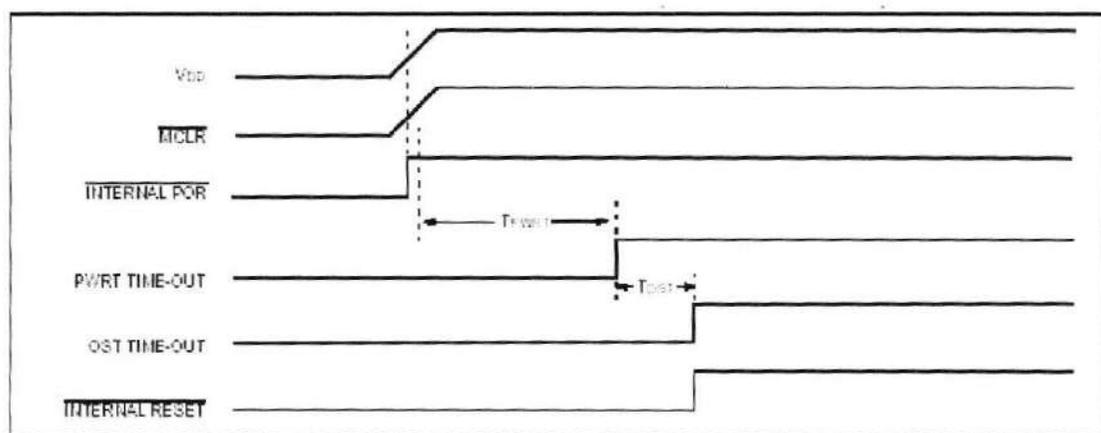


Figura 4.33.- Diagrama de tiempos de las señales que provocan un Reset.

La Tabla XXVI muestra los valores que toman los bits involucrados en un Reset:

Tabla XXVI.  
Indica las diferentes causas de Reset.

POR	BOR	TO	PD	Tipo de Reset
0	x	1	1	Conexión de alimentación (POR).
0	x	0	X	Illegal.
0	x	X	0	Illegal.
1	0	1	1	Por caída de tensión (BOR).
1	1	0	1	Por WDT (operación normal).
1	1	0	0	Por WDT (modo sleep).
1	1	U	U	Activación normal (MCLR#).
1	1	1	0	MCLR# en sleep o interrupción para despertar de sleep.

La x significa que su valor es uno o cero puede tomar cualquier valor y la u significa que su valor no varía con respecto al valor anterior al reset.

#### *Perro Guardián (WDT)*

El WDT es un contador que funciona con los impulsos del oscilador interno del dispositivo y provoca un Reset cuando se desborda en funcionamiento normal.

Si el desbordamiento se produce cuando el microcontrolador se halla en estado de Reposo, se despierta y sigue su comportamiento normal.

Las instrucciones CLRWDT y SLEEP borran o ponen a cero el valor de conteo del WDT y el del Postdivisor. Si se ejecuta la instrucción CLRWDT y el Predivisor de Frecuencia está asignado al perro guardián, se borra, pero no cambia su configuración.

La figura 4.34 muestra el diagrama de bloques del perro guardián:

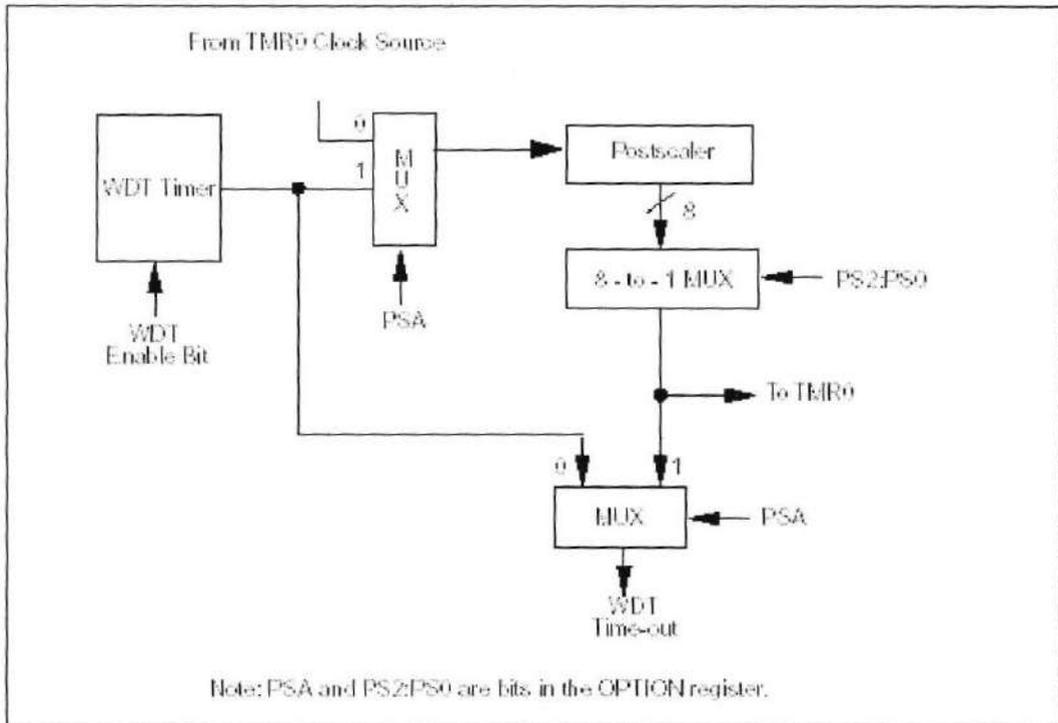


Figura 4.34.- Diagrama de Bloques del WDT.

#### *Modo Sleep (Reposo o de bajo consumo)*

Este es un modo en donde el dispositivo se encuentra en un estado de bajo consumo. El oscilador deja de funcionar por lo tanto ningún sistema que utiliza reloj funciona. El modo SLEEP se ejecuta con la instrucción SLEEP.

Si es habilitado, el WDT será limpiado su contador pero se mantendrá funcionando, el bit PD# del registro STATUS es limpiado, el bit TO# es puesto a uno, y el oscilador es apagado. Las puertas de E/S mantendrán su estado anterior al SLEEP, y su consumo será mínimo. Cuando la instrucción Sleep es ejecutada, la siguiente instrucción que se ejecuta cuando se despierta del Sleep es (PC+1) y las causas para despertar del SLEEP son:

- Activación externa del pin MCLR#.
- Desbordamiento del WDT que sigue trabajando en Reposo.
- Generación de una interrupción por activación del pin RB0/INT o por cambio en los 4 pines de más peso de la Puerta B.
- Interrupción por los periféricos internos anteriormente mencionados.

#### **4.2 INTERFACE DE COMUNICACION MAXIM-RS232A.**

##### *Norma RS-232*

El RS-232 permite la comunicación del microcontrolador con otros dispositivos inteligentes, como puede serlo el computador.

La norma de comunicación serie RS-232 es una de las más extendidas. Una de sus características es la de los rangos de tensiones que utiliza para representar los niveles lógicos. El nivel alto se representa con un nivel de voltaje comprendida entre -3V y -15V, mientras que el nivel bajo utiliza el rango de +3 V a +15V.

#### *Maxim-232A*

El Maxim-232A dispone de dos canales de entrada para niveles TTL, que son el  $T1_{IN}$  y el  $T2_{IN}$ , con sus correspondientes salidas,  $R1_{OUT}$  y  $R2_{OUT}$ . Igualmente, posee dos canales de entrada para niveles RS-232  $R1_{IN}$  y  $R2_{IN}$  y sus correspondientes salidas TTL,  $T1_{OUT}$  y  $T2_{OUT}$ . Se alimenta con +5V. Tiene dos convertidores de voltajes que convierte de +5 V a  $\pm 10$  V para la operación del RS-232. El primer convertidor usa un capacitor C1 para doblar la entrada de +5V a +10V en C3 en la salida V+. El segundo convertidor usa un capacitor C2 para invertir +10V a -10V en C4 en la salida V-.

Las entradas pueden ser compatibles con entradas TTL y CMOS. Las entradas de los drivers que no se estén utilizando pueden quedar desconectadas ya que hay una resistencia de  $400\text{k}\Omega$  unida a  $V_{cc}$ , llamadas Pull-up que obligan que su salida sea cero ya que dichos drivers invierten la entrada como se puede ver en la figura 4.35.

Los drivers de recepción reciben niveles de voltaje mayores que 3V como un cero lógico, que son invertidos por los drivers. Las entradas de los drivers de recepción aguantan sobrevoltaje arriba de los  $\pm 25$  y viene con una resistencia de entrada de  $5\text{k}\Omega$ .

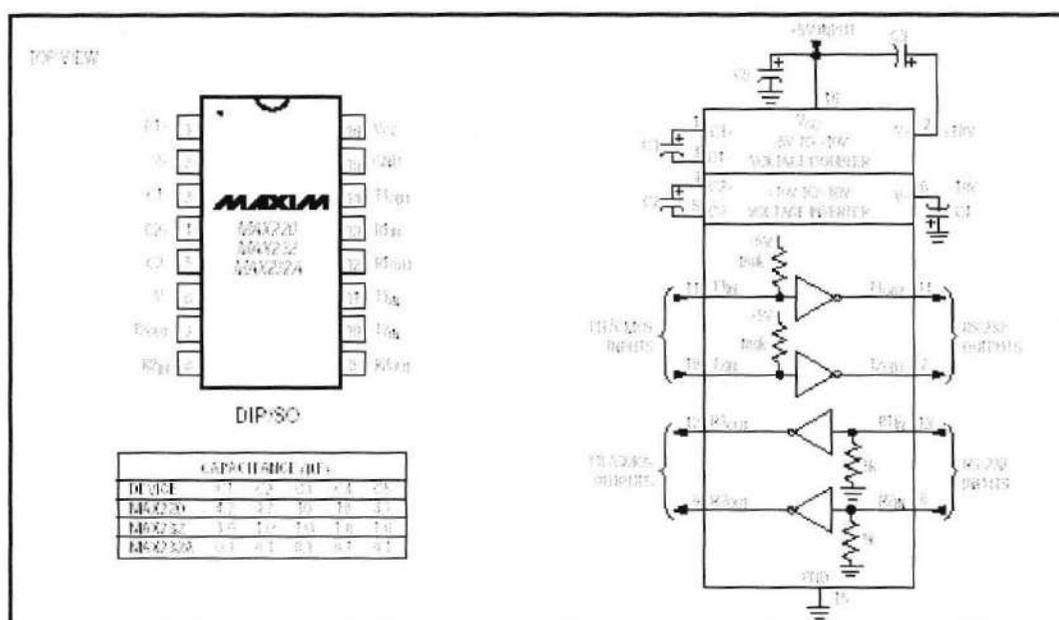


Figura 4.35.- Estructura interna del Maxim-RS-232A.

### 4.3 AMPLIFICADOR OPERACIONAL LM324.

#### Características

- Internamente frecuencia compensada por ganancia unitaria
- Gran ganancia de Voltaje DC 100 dB
- Amplio ancho de banda (ganancia Unitaria) 1MHz
- Temperatura compensada
- Amplio Rango de fuente de alimentación.



- Fuente de 3V a 32V o doble fuente  $\pm 1.5V$  a  $\pm 16V$
- Alimentación de corriente muy baja (700  $\mu A$ ) esencialmente Independiente de la fuente de voltaje
- Baja corriente Bias 45nA (Temperatura compensada)
- Baja entrada de Voltaje offset 2mV y corriente offset 5Na

#### Aplicaciones

La serie LM324 son opamps cuya operación es con solamente una sola fuente de voltaje. Estos amplificadores operan sobre un amplio rango de fuente de alimentación con un pequeño cambio en el rendimiento de las características. A 25°C la operación del amplificador es posible bajo una mínima fuente de alimentación de voltaje de 2.3VDC.

Los pines de salida del empaquetamiento han sido diseñados para simplificar los circuitos impresos. Las entradas inversoras están adyacentes a las salidas para todos los amplificadores y las salidas han sido además situadas en las esquinas del empaquetamiento (pines 1, 7, 8 y 14).

Deberían tomarse precauciones para asegurar que la fuente de alimentación de voltaje del circuito integrado nunca invierta su polaridad o que la unidad no sea inadvertidamente instalada al revés en un socket con lo que una sobrecorriente a través del diodo dentro del IC podría causar el derretimiento de los conductores internos y el resultado es la destrucción del dispositivo. Grandes entradas diferenciales de voltajes pueden ser fácilmente acomodadas y, como una protección de diodos para entradas diferenciales de corriente no son necesitadas, ninguna gran corriente de entrada resulta de ellas. La entrada diferencial de voltaje puede ser más grande que  $V+$  sin dañar el dispositivo. Se deberían proporcionar para prevenir las entradas de voltaje que provienen de entradas negativas mayores a  $-0.3VDC$  (a  $25^{\circ}C$ ). Una diodo con un resistor a la entrada de la terminal de entrada del IC puede ser usada.

Al reducir la fuente de poder drásticamente, el amplificador tiene un estado de salida Clase A para niveles pequeños de señales los cuales se convierten a una clase B en modos de señales grandes. Esto permite al amplificador ambos surtir y fugar grandes corrientes de salidas. Consecuentemente, para ambos NPN y

PNP transistores de ayuda pueden ser usados para extender la capacidad de la potencia del amplificador básico.

Para AC aplicaciones, donde la carga es capacitivamente acoplada a la salida del amplificador, un resistor debería ser usado, de la salida del amplificador a tierra para incrementar la corriente bias Clase A y prevenir distorsiones.

Donde la carga es directamente acoplada, como en aplicaciones DC, no hay distorsiones. Cargas capacitivas las cuales son aplicadas directamente a la salida del amplificador reducen el margen de la estabilidad del lazo. Valores de 50 pf pueden ser acomodados usando la conexión no-inversora de ganancia unitaria. Grandes ganancias de lazo cerrado o aislamientos resistivos deben ser usadas si grandes cargas capacitivas deben ser manejadas por el amplificador.

Las salidas de cortocircuito a tierra o al positivo de la fuente de alimentación deberían ser por un corto tiempo. Los dispositivos pueden destruirse, no como un resultado de una corriente de cortocircuito cause el derretimiento del metal, más bien debido a un gran incremento de la disipación en el chip la cual causará una eventual falla debido a excesivas temperaturas de juntas. Colocando



cortocircuitos en más de una salida a la vez incrementará la potencia disipada total a niveles destructores, si no es propiamente protegidos con externos resistores de disipación limitantes en serie con la salida de los amplificadores. El gran valor de corriente de alimentación el cual es disponible a 25°C provee una gran capacidad de corriente de salida en elevadas temperaturas que un opamp estándar.

#### *Descripción*

La serie LM324 consiste de cuatro independientes amplificadores operacionales con alta ganancia, frecuencia compensada internamente quienes fueron diseñados específicamente para operar de una sola fuente de alimentación sobre un amplio rango de voltajes. La operación de varias fuentes de alimentación divididas es además posible y la baja corriente de alimentación es independiente de la magnitud de la fuente de alimentación de voltaje.

Las áreas de aplicación incluyen amplificadores transductores, bloques de ganancia DC y los circuitos convencionales de opamps los cuales pueden ser

más fácilmente implementados con simples sistemas de fuentes de alimentación.

La Figura 4.36 nos muestra la configuración de los cuatro amplificadores.

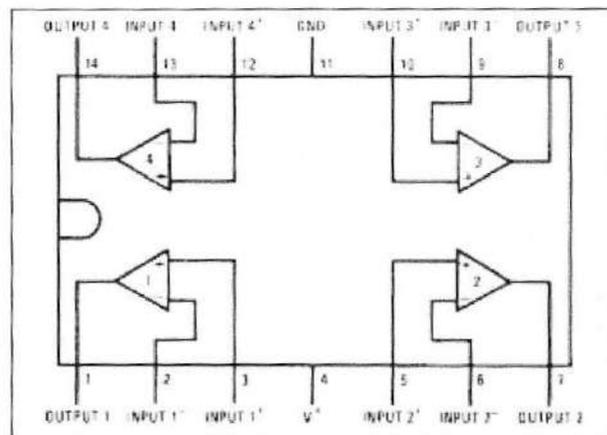


Figura 4.36.- Muestra la asignación de pines del LM324.

#### 4.4 DIODO ZENER LM336.

##### *Características*

- Bajo coeficiente de temperatura
- Amplia Corriente de operación de 400  $\mu$ A a 10 mA

0.2 W de impedancia dinámica

- $\pm 1\%$  tolerancia inicial disponible

-Estabilidad de temperatura garantizada

-Fácilmente ajustable para un mínima temperatura propensa

-Rápido apagado

-Empaquetamiento de tres pines

### *Aplicaciones*

La serie LM336 proporciona un voltaje de referencia que es mucho más fácil de usar que un ordinario diodo zener. Su baja impedancia y amplio rango de corriente de operación simplificando la polarización en casi cualquier circuito. Además, cualquiera, el voltaje de referencia o el coeficiente de temperatura pueden ser ajustados para optimizar el rendimiento del circuito.

La Figura 4.37 muestra un LM336 con un potenciómetro de 10K para ajustar el voltaje de referencia. Con la adición de R1, el voltaje de referencia puede ser ajustado sin afectar el coeficiente de temperatura de él dispositivo. El rango de



ajuste es usualmente suficiente para ajustar ambos la tolerancia inicial del dispositivo y la inexactitud en circuiterías intermediarias.

Si el coeficiente mínimo de temperatura es deseado, dos diodos pueden ser adicionados en serie con el potenciómetro de ajuste. Casi cualquier diodo puede ser usado para este propósito tal como un 1N914, 1N4148 o un 1N457. Para una compensación apropiada de temperatura los diodos deberían estar en el mismo ambiente térmico que el LM336. Es usualmente suficiente montar los diodos cerca del LM336 en el circuito impreso. La resistencia absoluta de R1 no es crítica y cualquier valor entre 2K a 20K trabajará.

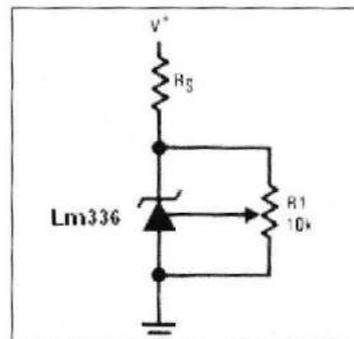


Figura 4.37.- Muestra la configuración para ajustar el voltaje de ruptura y coeficiente de Temperatura del LM336.

### *Descripción*

El circuito integrado LM336 es un diodo regulador de precisión de 5V. Opera como un Zener de coeficiente bajo de temperatura con 0.2 W de impedancia dinámica. Un tercer terminal en el LM336 permite que el voltaje de referencia y el coeficiente de temperatura sean ajustables fácilmente.

La serie LM336 es conveniente como un voltaje de referencia de 5V para voltímetros digitales, fuentes de poder o circuitería donde se ven involucrados opamps. Adicionalmente, el Lm336 puede ser usado como un positivo o negativo voltaje de referencia. El LM336 tiene un rango de operación de 0°C a 70°C.

La Figura 4.38 muestra la configuración interna del empaquetamiento tipo transistor.

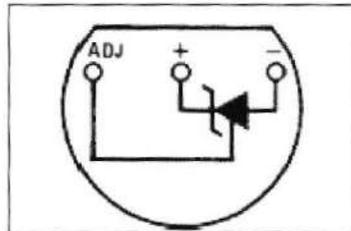


Figura 4.38.- Muestra la asignación de pines del LM336.

#### 4.5 OPTOAISLADOR 4N35.

##### Características

- 3500, 2500 o 1500 Voltios de aislamiento.
- Alta transferencia de Corriente DC.
- Bajo Costo.
- Interfase con familias lógicas comunes.
- Entradas-Salidas con capacitancia de acoplamiento  $<0.5$  pF.

## Aplicaciones

Detecciones de redes AC.

Manejo de Relays.

Fuentes de switch.

Detección de llamadas telefónicas.

Aislamiento de tierra lógica.

Acoplamiento lógico con ruido de alta frecuencia.

## *Descripción*

El optoaislador 4n35 es un acoplador-aislador óptico que consiste de diodos emisores infrarrojos de Galio arseniuro y un fototransistor NPN montado en un empaquetamiento común de 6 pines.

Los optoaisladores han sido probados su aislamiento con un voltaje 5300 VRMS. El funcionamiento del aislamiento es completado a través de procesos de fabricación de aislamientos regidos bajo normas ISO9001 resultando en un

alto funcionamiento de aislador. El circuito de la Figura 4.39 muestra la configuración interna del optoaislador.

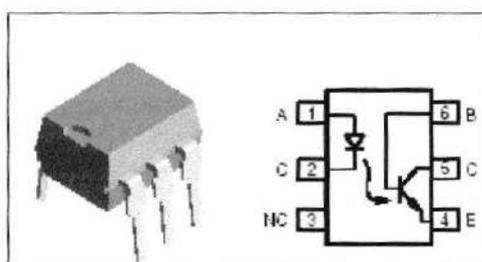


Figura 4.39.- Muestra la configuración interna del optoaislador 4N35, consta de un diodo emisor y un fototransistor.

#### 4.6 SENSORES.

En este sistema de monitoreo se usan tres sensores para los medir los tres parámetros más importantes del sistema, ya que de esos tres parámetros se derivan muchos más.



CIB-ESPOL

#### 4.6.1 TEMPERATURA.

El dispositivo LM334 es una fuente ajustable de corriente de tres terminales, puede servir usado como una excelente regulación de corriente y se lo puede alimentar con un rango amplio de voltaje de 1 V a 40 V. El LM334 de la Figura 4.40 establece su corriente usando un resistor externo, con una exactitud de  $\pm 3\%$ . Además aplicando voltajes revertidos por arriba de los 20V extraerá solamente pocas docenas de microamperios de corrientes, permitiendo actuar como un rectificador y una fuente de corriente en aplicaciones AC.

El voltaje medido para establecer la corriente de operación en el LM334 es de 64 mV a 25°C y es directamente proporcional a la temperatura en °K. Con una simple conexión de un resistor externo se genera una corriente con un 0.33%/°C de dependencia con la temperatura.

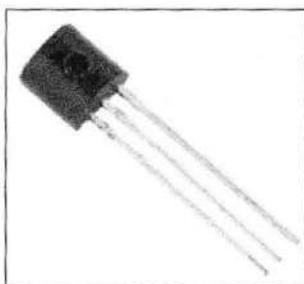


Figura 4.40.- Sensor de Temperatura LM334.

Se usa en aplicaciones en donde las fuentes de corriente incluyen protecciones de sobrecorriente, referencias de baja potencia, generación de rampas, controladores de led, y medición de temperatura el LM334 es recomendable ser utilizado. Por lo general el LM334 es ideal para remotas mediciones de temperatura donde hay largas distancias entre el sensor y el procesamiento de la señal, ya que la exactitud no es alterada, mientras que sensores donde la señal que emite el sensor es un voltaje, con largas distancias provocaría inexactitud. Además con el LM334 sólo requiere 2 cables.

El LM334 mide voltajes desde 0°C a 70°C. Las características del sensor se enumeran a continuación:

- Opera con una alimentación de 1V a 40V.
- Regulación de corriente 0,02%/V.
- Programable de 1  $\mu$ A a 10 mA.
- Operación con dos terminales.
- Disponible como un sensor de temperatura.
- $\pm 3\%$  de exactitud.

Para el funcionamiento del LM334 requiere de una resistencia externa, la cual debe se calculada de la siguiente manera:

#### *Cálculo de Reset*

La corriente total a través del LM334, ( $I_{SET}$ ), es la suma de la corriente que va a través de  $R_{SET}$  ( $I_R$ ) y la corriente de BIAS ( $I_{BIAS}$ ), como se ve en la figura 4.41:

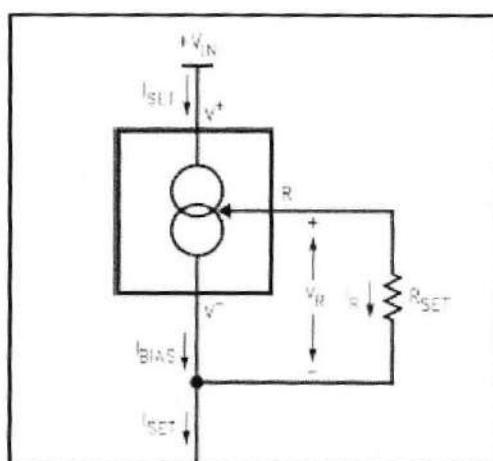


Figura 4.41.- Diagrama de conexiones.

La corriente que fluye a través de RSET es determinada por VR, la cual es aproximadamente  $214 \mu\text{A}/^\circ\text{K}$  ( $64\text{mV}/298^\circ\text{K}$ ) según la ecuación 4.5.

$$I_{\text{SET}} = I_{\text{R}} + I_{\text{BIAS}} = \frac{V_{\text{R}}}{R_{\text{SET}}} + I_{\text{BIAS}}$$

Ecuación 4.5.-

La corriente  $I_{\text{BIAS}}$  simplemente un porcentaje de  $I_{\text{SET}}$ , la ecuación 4.5 puede ser re-escrita como:

$$I_{\text{SET}} = \left( \frac{V_{\text{R}}}{R_{\text{SET}}} \right) \left( \frac{n}{n-1} \right)$$

Ecuación 4.6.-

Donde n es el ratio de  $I_{\text{SET}}$  a  $I_{\text{BIAS}}$ . Puesto que n es típicamente 18 para  $2\mu\text{A} \leq I_{\text{SET}} \leq 1\text{mA}$ , la ecuación 4.6 puede ser simplificado a:

$$I_{\text{SET}} = \left( \frac{V_{\text{R}}}{R_{\text{SET}}} \right) (1.059) = \frac{227 \mu\text{V}/^\circ\text{K}}{R_{\text{SET}}}$$

Ecuación 4.7.-

### *Efectos Térmicos*

El calentamiento interno puede ser un efecto significativo en regulaciones de corriente para  $I_{SET}$  mayores de 100  $\mu A$ . Por ejemplo, por cada incremento de 1V que se incrementa a través del LM334 en  $I_{SET} = 1$  mA se incrementará la temperatura de unión igual a 0.4 °C. La corriente  $I_{SET}$  tiene un coeficiente de temperatura de 0.33 %/°C, así el cambio en la corriente debido al crecimiento de temperatura será  $(0.4) (0.33) = 0.132\%$  La relación es 10:1 de degradación en regulación comparado con los verdaderos efectos eléctricos, por lo tanto no podremos tener una buena regulación de la corriente o en aplicaciones de mediciones de temperatura, las mediciones tendrán una elevada inexactitud.

### *Ruido*

El ruido generado por el LM334 es aproximadamente 4 veces el ruido de disparo de un transistor. Si el LM334 es usado como una carga activa por un transistor amplificador, el ruido referido de entrada será incrementado aproximadamente en 12 dB.

### *Resistencia de Contacto*

Los voltajes medidos que determinan la corriente de operación del LM334 son menos de 100mV. En estas condiciones, los efectos resistivos podrían ser minimizados localizando la resistencia  $R_{SET}$  físicamente cerca del dispositivo o sensor. De ser posible un sockets o enchufes deben ser evitados de ser posible. Demanda solamente 0.7W la resistencia de contacto para reducir la corriente de salida en un 1% a un nivel de 1mA.

### *Midiendo Temperatura*

El LM334 hace ideal las mediciones de temperatura remotas porque su modo de operación de corriente no hace perder exactitud a lo que fluye en largos cables.



CIB-ESPOL

La salida de Corriente es directamente proporcional a la temperatura absoluta ( $^{\circ}K$ ), de acuerdo con la siguiente fórmula (Ecuación 4.8):

$$I_{SET} = \frac{(227 \mu V/^{\circ}K) (T)}{R_{SET}}$$

Ecuación 4.8.-

Por lo tanto tenemos una corriente proporcional a la temperatura que estamos midiendo, luego conectamos una resistencia a la salida del LM334 y obtenemos un voltaje que lo podemos medir y hacer la relación con la temperatura.  $R_{SET}$  como ya se dijo debe estar conectado cerca del sensor en aplicaciones donde hay mucha distancia entre el punto de medición y el tratamiento del dato, en la figura 4.42 se puede observar una típica configuración.

Con esta aplicación sólo se necesita dos hilos de cable, que preferentemente debe ser cable apantallado para evitar el ruido que genera el propio sensor y los dispositivos electrónicos que estén cerca.

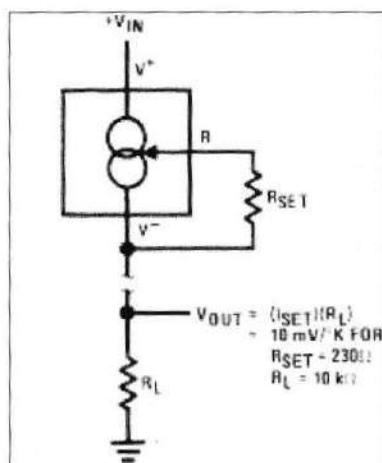


Figura 4.42.- Típica conexión para mediciones remotas de temperatura con dos hilos de cable.

#### 4.6.2 CORRIENTE

El sensor utilizado para medir la corriente es el "SHUNT" no es nada más un pedazo de alambre o barra de cobre con determinado resistencia, muy pequeña, que me produce una relación con la cual puedo determinar la corriente, la relación es la ley de Ohm, ya que al pasar una corriente por el shunt, tenemos un voltaje proporcional, a la corriente que pasa por la barra de cobre.} Hay diferentes tipos de shunt, y que los observamos en los cargadores de batería, para medir la corriente DC que circula por las baterías por ejemplo, hay shunts de 100mV por 30A, o por 15A, todo depende de su valor de resistencia.

### 4.6.3 VOLTAJE.

El voltaje de las baterías va creciendo a medida que reciben carga. La carga recibida tiene un máximo de 15 Amperios. Para poder medir el voltaje se usa una configuración con dos resistencias formando un divisor de voltaje conectadas de tal manera que reducimos el valor de voltaje de la línea de baterías, situada aproximadamente en 200 Voltios, a un valor dentro de los rangos TTL proporcional al Voltaje real así podemos introducirlo a la tarjeta esclavo para ser procesado. La ecuación 4.9 indica la señal de Voltaje proporcional al Voltaje de la batería, vemos que al aumentar el valor de R1 reducimos la señal del voltaje que se introduce a la tarjeta.

$$\text{Señal Voltaje} = \left( \frac{R2}{R1 + R2} \right) \times VBateria$$

Ecuación 4.9.-

Se debe escoger el valor de este par de resistencias según el procesamiento de la información en el Firmware Observamos en la Figura 4.43 que la

resistencia R1 es de mayor valor debido a que así la corriente que circula por esta parte del circuito es en el orden de los microamperios mientras que la corriente circulante por las baterías es de máximo 15 Amperios como ya se mencionó, protegiendo al circuito de la tarjeta esclavo de sobrecorrientes.

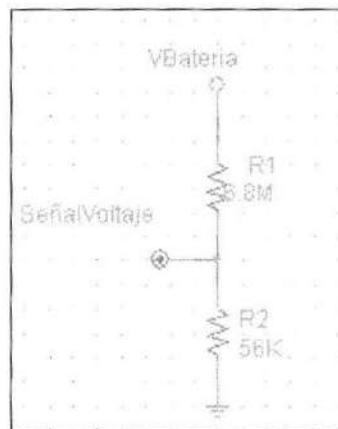


Figura 4.43.- Divisor de Voltaje usado para medir el valor del voltaje de las baterías.

La figura 4.43 corrobora la relación en la fórmula 3.4 del capítulo 3 con la que el software determina el voltaje en la línea de baterías.

# **CAPITULO 5**

## **5.-DISEÑO DEL HARDWARE Y EL FIRMWARE.**

El sistema de Monitoreo desarrollado consta de un hardware que desarrolla un sistema de tratamiento de señales analógicas que provienen de los datos de las baterías y usa como procesador central de datos un Microcontrolador PIC 16F877A el cual además de procesarlos por medio del Firmware que lleva en su interior, realiza una comunicación serial I2C en circuito cerrado con otros microcontroladores, luego el microcontrolador central se comunica con la PC y

el software "Sistema de Monitoreo de Baterías", que posteriormente analiza los datos.

### **5.1 INTERCONEXION DEL SISTEMA Y FUNCIONAMIENTO DEL FIRMWARE.**

El sistema de monitoreo en este proyecto, como ya se menciono es un sistema Maestro-Esclavo, donde todas las tarjetas esclavas recogen los datos de los cargadores y lo envían a la tarjeta Maestro que organiza los datos en un orden específico para luego enviarlos a la PC (Host-Computer), en la figura 5.1 podemos observar la interconexión del sistema.

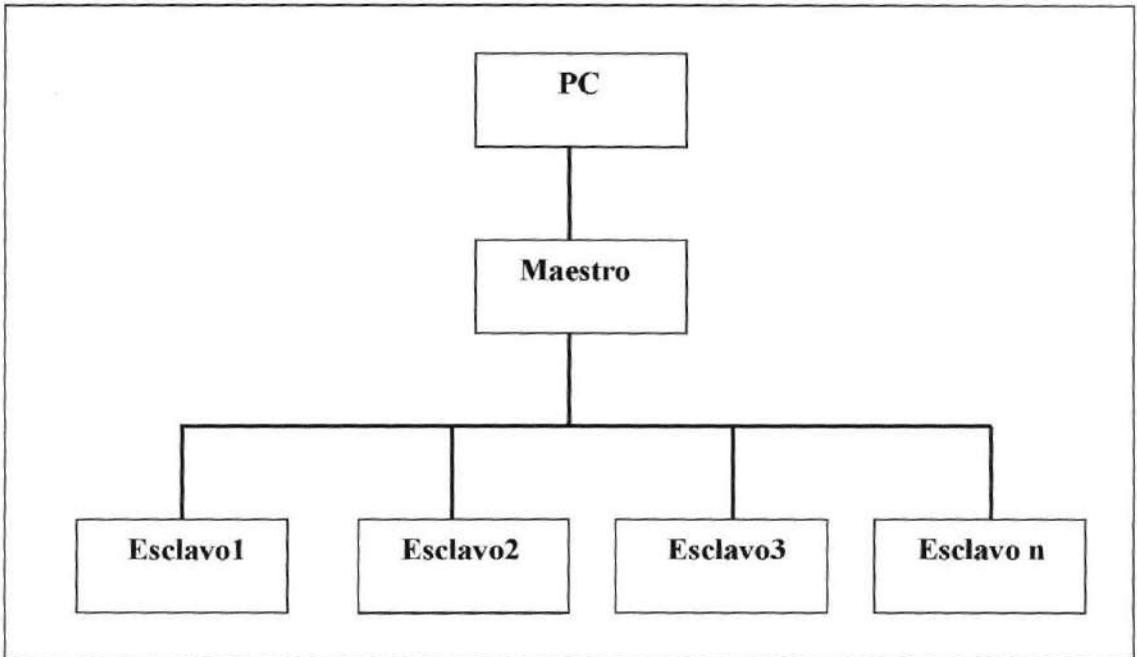


Figura 5.1.- Interconexión del sistema.

En el sistema de monitoreo, el maestro direcciona a cada esclavo, simulando una llamada telefónica, es decir que cada esclavo tiene una dirección determinada a la cual responde al llamado del maestro. El tiempo entre un direccionamiento y otro se denomina tiempo de monitoreo, este lo podemos establecer en el software de monitoreo del Capítulo 3.

La dirección de cada esclavo es un número de 8 bits, esta dirección se establece en el Firmware del Pic esclavo, pero estas direcciones según el

modelo I2C son números impares, como es un número de 8 bits, el número de direcciones es de 128 pero en la práctica hay algunas direcciones que están reservadas, para cierto tipo de comunicaciones en el mismo formato I2C, así que el número de esclavos máximo teórico es de 128, pero en nuestro proyecto direccionamos dos módulos esclavos.

En el módulo I2C hay modos de direcciones de 10 bits, con lo que aumentaría el número de esclavos a 512 aproximadamente, para esto hay que configurar algunos registros de control del Firmware del Pic.

Como este sistema recolecta datos de diferentes líneas de carga y cada una tiene circuitos de fuerza y referencias diferentes, por lo tanto no se pueden poner a la misma referencia a todos, así que el sistema debe recoger datos uno a uno. Por lo que el sistema de monitoreo (PC-Maestro) introduce una tarjeta esclavo montada en un cargador al sistema y recoge sus datos, manteniendo a las demás tarjetas esclavos aislados del sistema pero recolectando datos, hasta que le toque su turno a cada uno de ingresar al sistema.

### *Funcionamiento*

El Firmware se encuentra en la memoria del programa del Pic, e interactúa con los demás dispositivos que se encuentran en la tarjeta electrónica, hay varios módulos principales unos propios del Pic y otros que se han desarrollado en este proyecto que trabajan conjuntamente para obtener los datos de carga de la batería.

Estos módulos se pueden ver como trabajan con un diagrama de bloques de la figura 5.2 y 5.3:



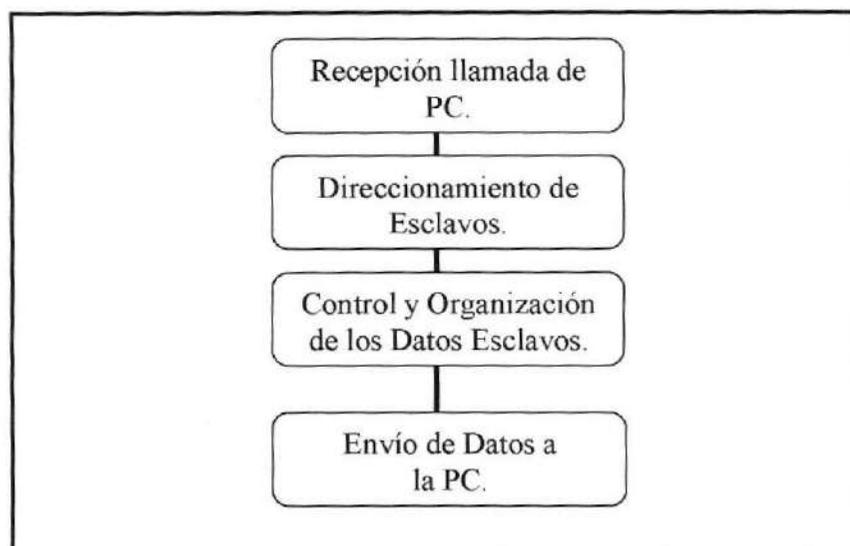


Figura.5.2- Diagrama de Bloques de la tarjeta maestro.

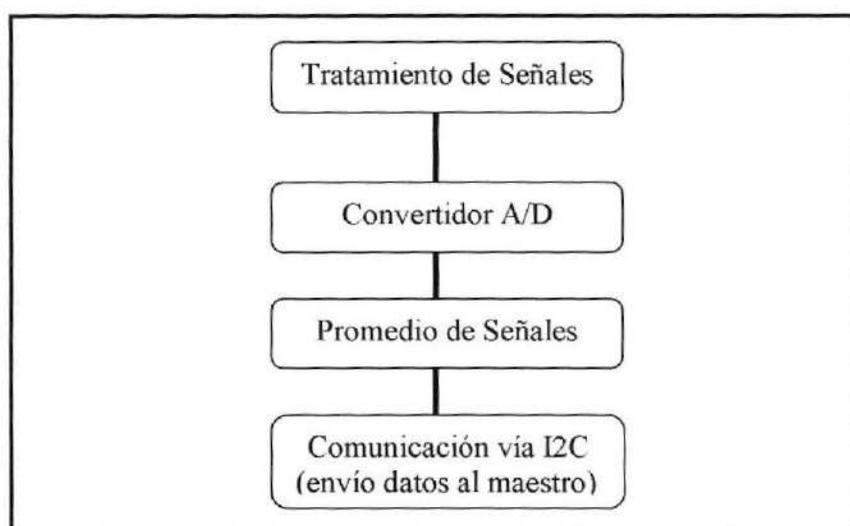


Figura 5.3.- Diagrama de Bloques de las tarjetas esclavo.

Como ya se mencionó anteriormente el sistema de monitoreo consta de dos partes principales que son los módulos esclavos y el módulo maestro, responsables de la recolección de datos

## **5.2 MODULO MAESTRO**

El módulo Maestro consta de una sola tarjeta central donde llegan todos los datos de los módulos Esclavo, su función específica es ordenar los datos de carga de las líneas de baterías que se encuentran monitoreando los módulos Esclavos y enviarlos a la PC. El módulo Maestro tiene varios módulos internos que son manejados por un Firmware al que llamaremos "Firmware Maestro". El Firmware Maestro maneja la comunicación I2C con los módulos Esclavos, pasa al módulo de control de memoria EEPROM para organizar los datos y guardarlos en las localidades adecuadas de la RAM e EEPROM. Por último con los datos organizados por esclavo, el Firmware maneja el módulo USART para comunicarse con la PC y enviarle los datos para que el Software "Sistema de Monitoreo" los analice.

### **5.2.1 MODULO DE COMUNICACION I2C.**

El sistema de monitoreo desarrolla un sistema de comunicación bajo las especificaciones del sistema I2C, y para mantener el buen rendimiento de este sistema, se debe reunir un adecuado ambiente de monitoreo, así si un dato tiende a ser una falla pueda ser identificado rápidamente, así mismo se buscó tener un sistema a bajos costos. Buscando rapidez e incrementos de la flexibilidad de este sistema de comunicación se usa una característica modular que recoge los datos de cada línea de baterías y por la proximidad de cada uno de ellos se usa el bus I2C el cual es una solución por la mínima cantidad de Hardware y costos. En el sistema están conectados varios Microcontroladores que a su vez tiene una amplia gama de periféricos, los cuales se pueden ir conectando al bus I2C en una tarjeta llamada módulo esclavo, teniendo las mismas características cada tarjeta y sin necesidad de ninguna configuración adicional en el módulo Maestro que está conectado al bus I2C, que reúne los datos de todos los módulos esclavos organizándolos y enviándolos a una PC, para su posterior análisis. Para recolectar los datos de una buena forma y con un alto rendimiento, se ejecutó un protocolo que previene al sistema de errores que se analizarán en las secciones siguientes.

Los registros que se utilizan son:

- a) SSPCON y SSPCON2 que son de control de la transmisión.
- b) SSPSTAT registro del estado de la última transferencia.
- c) SSPBUF donde se escribe el dato que se va a transmitir.
- d) SSPADD donde se guarda la dirección del Esclavo.

A continuación se muestra los bits de los registros:

### REGISTRO DE ESTADO SSPSTAT.

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D $\bar{A}$	P	S	$\bar{RW}$	UA	BF
bit 7							bit 0

**SMP:** Bit de muestreo.

**1** = Control del muestreo es deshabilitado para modos de velocidad Standard (100KHz y 1 MHz).

**0** = Control del muestreo es habilitado para modos de alta velocidad (400 Khz.).

**CKE:** Selección de niveles de entrada.

**1** = Niveles de entrada conforme al bus SMBUS.

**0** = Niveles de entrada conforme al bus I2C.

**D/A:** Bit Datos/Dirección.

**1** = Indica que el ultimo dato recibido o transmitido fue un dato.

**0** = Indica que el ultimo dato recibido o transmitido fue una dirección.

**P:** Bit de Parada.

**1** = Indica que un bit de parada ha sido recibido.

**0** = El bit de parada no fue detectado.

**S:** Bit de inicio.

**1** = Indica que un bit de parada ha sido recibido.

**0** = El bit de parada no fue detectado.

**R/W:** Bit de información Lectura/Escritura.

Modo Esclavo:

**1** = Lectura.

**0** = Escritura.

Modo Maestro:

**1** = Transmisión en progreso.

**0** = Transmisión no en progreso.

**UA:** Bit de modo de dirección de 10 bits.

**1** = Indica que la dirección es de 10 bits.

**0** = Indica que la dirección es de 7 bits.

**BF:** Bit de Buffer lleno.

Receptor:

**1** = Recepción completa, Buffer lleno.

**0** = Recepción incompleta buffer vacío.

Transmisor:

**1** = Transmisión de dato en progreso, SSPBUF lleno.

**0** = Transmisión de dato incompleta, SSPBUF vacío.

### REGISTRO CONTROL SSPCON

R/W-0							
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
bit 7							bit 0

**WCOL:** Bit de detección de Colisión en escritura.

Modo Maestro:

**1** = Una escritura a SSPBUF fue intentada mientras la condiciones del bus I2C no eran válidas.

**0** = No hay colisión.

Modo Esclavo:

**1** = SSPBUF es escrito mientras hay una transmisión previa en progreso.

**0** = No hay colisión.

**SSPOV:** Bit indicador de desbordamiento.

**1** = Un byte es recibido mientras SSPBUF tiene un byte y no ha sido leído.

**0** = No hay desbordamiento.

**SSPEN:** Bit habilitación del Puerto serie. .

**1** = Habilita el puerto serial y configure SDA y SCL como líneas del Puerto serial.

**0** = Deshabilita el Puerto serial y configura estos pines como puertas de Entrada/Salida.

**CKP:** Bit de habilitación del reloj.

**1** = Habilita el reloj.

**0** =Mantiene el reloj en bajo.

**SSPM3:SSPM0:** Bits de selección del modo del puerto serial.

### REGISTRO CONTROL SSPCON2

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
bit 7							bit 0

**GCEN:** Bit de habilitación de llamada General (Solo modo Esclavo).

**1** = Se habilita una interrupción cuando una llamada general (0000h) es recibida.

**0** = La llamada General es deshabilitada.

**ACKSTAT:** Bit de estado de reconocimiento ACK (Solo en modo Maestro transmisor)

**1** = Bit ACK no fue recibido del esclavo.

**0** = Bit ACK fue recibido del esclavo.

**ACKDT:** Bit de reconocimiento ACK (Solo modo Maestro receptor)

Valor que será transmitido cuando el usuario inicia una secuencia de reconocimiento al final de la recepción.

**1** = No ACK.

**0** = ACK.

**ACKEN:** Bit de Habilitación de secuencia de reconocimiento (En modo Maestro receptor solamente).

**1** = Inicia la secuencia de reconocimiento ACK en SDA y SCL y transmite el bit ACKDT.

**0** = Secuencia de reconocimiento inactivo.

**RCEN:** Bit de Recepción habilitada (En modo maestro solamente)

**1** = Habilita el modo de recepción I2C.

**0** = Recepción inactivo.



**PEN:** Bit de condición de Parada habilitada (En modo maestro solamente)

**1** = Se inicia la condición de Parada en SDA y SCL.

**0** = Condición de Parada inactiva.

**RSEN:** Bit de condición de repetición de inicio habilitado (En modo maestro solamente).

**1** = Condición de repetición de inicio en SDA y SCL.

**0** = Condición de repetición de inicio inactiva.

**SEN:** Bit de habilitación de condición de inicio (En modo maestro solamente)

**1** = Condición de inicio en SDA y SCL.

**0** = Condición de inicio inactiva.

El bus I2C tiene dos líneas una que proporciona los pulsos del reloj y la otra línea bidireccional de datos, dichas líneas se ponen en un alto a través de un resistor conectado a Vcc como se ve en la figura 5.4. El módulo maestro direcciona a los módulos esclavos, en este sistema se usa el direccionamiento de 7 bits lo que permite en teoría 128 módulos esclavos conectados al bus I2C pero como ciertas direcciones están reservadas, físicamente se pueden conectar 112 módulos esclavos, pero para el proyecto se implementan dos módulos esclavos. Las líneas utilizadas son SDA (Datos) y SCL (Reloj), cada cambio en la línea SDA debe hacerse con la línea SCL en un bajo.

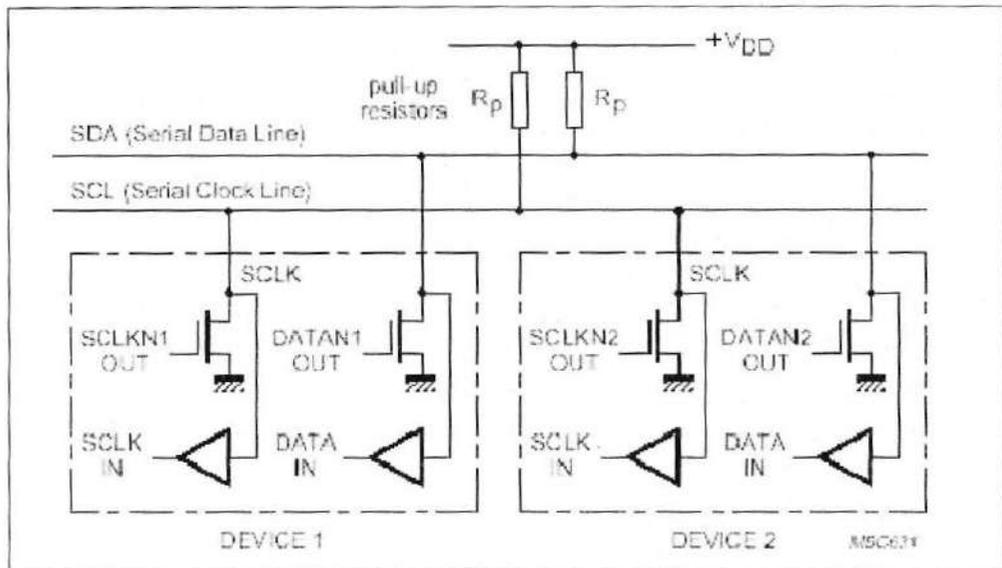


Figura 5.4.- Conexión de varios dispositivos al bus I2C.

### *Transferencia del Bit*

Como existe una gran variedad de tecnologías empleadas en los dispositivos para conectarse al bus I2C (CMOS, NMOS, TTL, etc.), los niveles lógicos "0" y "1" de los bits transferidos no tienen una tensión fija, sino que dependen de la tensión Vdd de alimentación.

### Validez del bit

La línea SDA es la encargada de tomar el valor correspondiente para cada uno de bits que constituyen un byte. Cada bit que se transfiere es acompañado por un pulso de SCL. Los valores en la línea SDA son válidos y deben permanecer estables durante el periodo alto de la señal de reloj (SCL).

El valor de la línea SDA puede variar cuando la señal de reloj (SCL) está en estado bajo (como se ve en la figura 5.5).

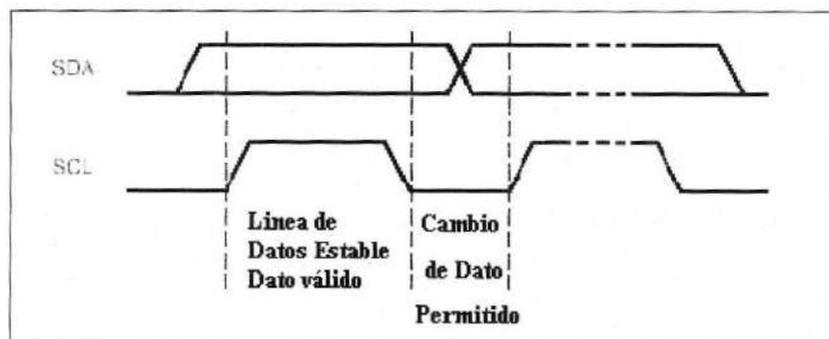


Figura 5.5.- Validez del Dato.

Los diferentes eventos o condiciones que se utilizan en la comunicación I2C de este proyecto se explican a continuación:

#### *Condición de Inicio*

Para iniciar una transferencia hay que determinar la condición de inicio mediante el Bit START, está formado por una transición de alto a bajo en la línea SDA mientras la línea SCL está a nivel alto (Figura 5.6). (No puede confundirse con un bit). Para generar esta condición, se escribe el bit SEN = 1 en SSPCON2. Si SDA y SCL se mantienen a nivel alto, el contador BRG se carga con el valor de 7 bits de SSPADD. Cuando termina el conteo de BRG la línea SDA pasa a nivel bajo. La condición consiste en un flanco descendente de SDA mientras SCL permanece con nivel alto. Pone el bit S = 1 del registro SSPSTAT. Luego se recarga el BRG con el valor SSPADD <6-0> y vuelve a decrementarse. Al llegar a 0 el contador, el bit SEN pasa a cero automáticamente.

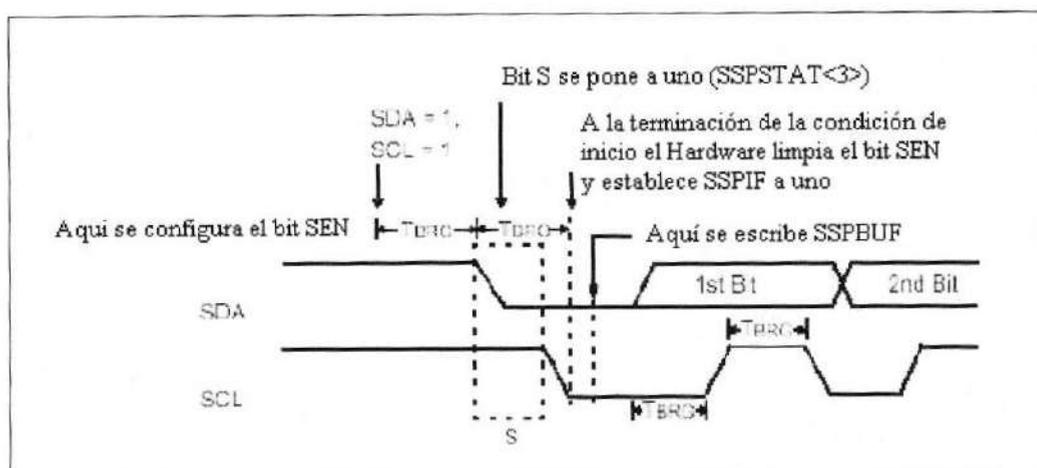


Figura 5.6.- Generación de la condición de Inicio.

#### Secuencia de Reconocimiento.-

El bit de reconocimiento ACK es obligatorio en la transferencia de cada byte. El pulso de reloj del bit N° 9 lo genera el maestro. El transmisor pone la línea SDA en alto durante dicho pulso de reloj. El receptor, por su parte, pone en bajo la línea SDA durante el pulso de reloj correspondiente al ACK y lo mantiene estable durante todo el período de dicho pulso. En la línea SDA prevalece por tanto el nivel bajo como se ve en la figura 5.7. Normalmente, el receptor que ha sido direccionado está obligado a generar el bit ACK después de recibir cada byte. Cuando un receptor esclavo no genera el bit ACK (NACK) al ser direccionado, debe mantener la línea SDA en alto durante el bit ACK (NACK).

Esta situación es detectada por el maestro, que debe generar la condición de parada y abortar así la transferencia.

Si un maestro está recibiendo datos debe generar ACK tras cada byte enviado por el esclavo excepto en el último, en el que el bit ACK lo mantiene en alto (NACK) y genera la condición de parada.

Para producir una secuencia de reconocimiento se pone el bit ACKEN = 1, entonces la patita SCL pasa a nivel bajo y aparece en SDA el bit de reconocimiento al finalizar el dato. Al poner ACKEN = 1 y ACKDT = 0 la línea SCL queda en bajo, mientras que el generador de frecuencia cuenta un período (TBRG). Luego SCL pasa a nivel alto y al completarse otro período SCL vuelve a pasar a nivel bajo y ACKEN = 0. El indicador SSPIF = 1 cuando finaliza la secuencia de reconocimiento (Figura 5.7).

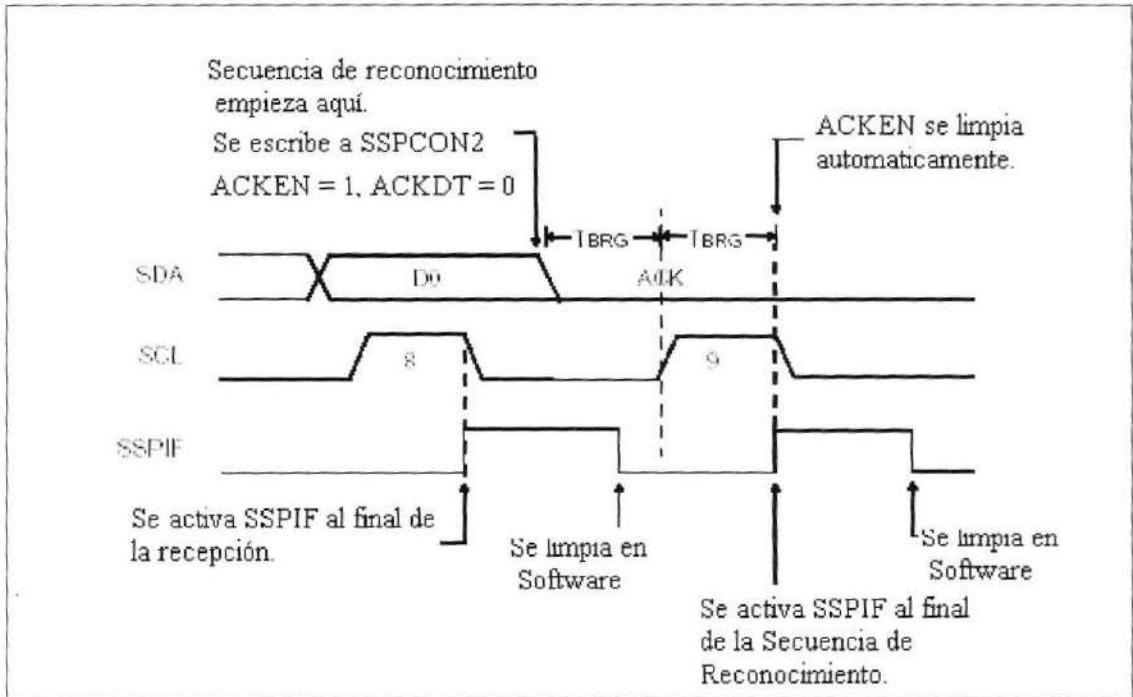


Figura 5.7.- Generación de la Secuencia de Reconocimiento.

### Condición de Parada

Se genera la condición de parada al final de la transmisión, poniendo el bit  $PEN = 1$  ( $SSPSON2<2>$ ). En el flanco descendente del noveno impulso de reloj la línea SCL queda en bajo y el generador cuenta un período, tras el cual SCL pasa a nivel alto. Tras otro período TBRG la línea SDA pasa a nivel alto y  $P = 1$  ( $SSPSTAT<4>$ ). Tras otro período PEN, se pone a 0 automáticamente.

El impulso de reloj en SCL se produce cuando se muestrea dicha línea cada ciclo  $T_{osc} * 4$ , estando a nivel alto, en cuyo momento se carga el contador BRG con los 7 bits de menos peso de SSPADD y comienza a contar otro período, pasando SCL a un bajo, al completar el período. En la figura 5.8 se encuentra la condición de Parada.

Ambas condiciones inicio y parada son siempre generadas por el maestro que, en todo momento, inicia y finaliza toda transferencia. El bus se considera ocupado tras la condición de inicio y, se considera libre, cierto tiempo después de la condición de parada.

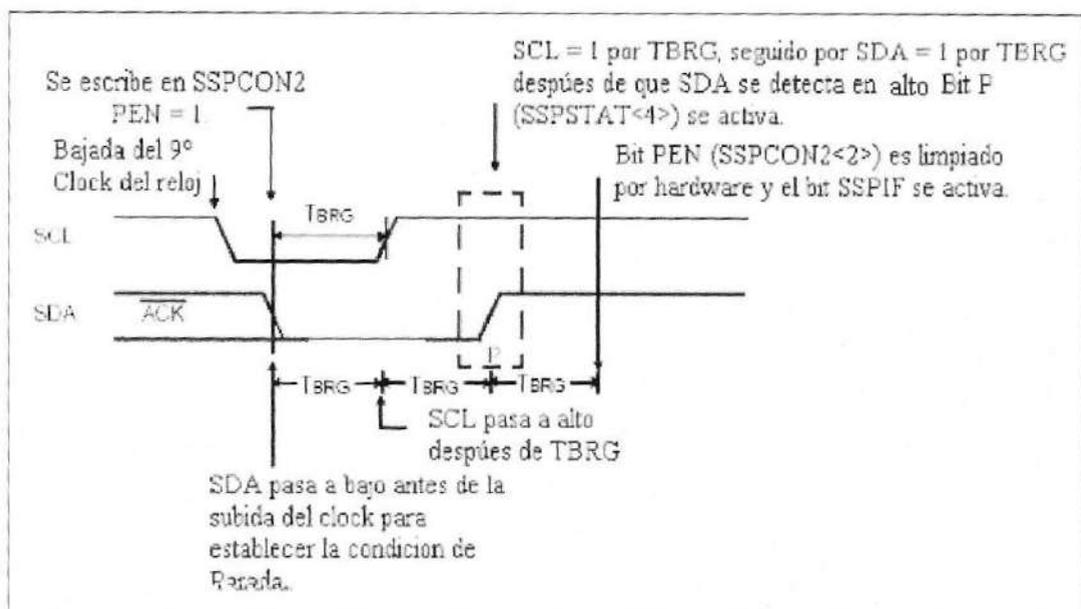


Figura 5.8.- Generación de condición de Parada.

### Formato de la Transmisión

El formato de la transmisión usada en el sistema de monitoreo se encuentra en la figura 5.9. Tras la condición de inicio, el maestro envía la dirección al esclavo al que se quiere dirigir. La dirección es de 7 bits más el bit RW que indica si la transmisión es de escritura o lectura, en el caso particular del sistema de monitoreo de este proyecto esta transmisión es de lectura (RW = 1). Luego de direccionar al esclavo, este envía un ACK y luego el número de datos que va a enviar al maestro, por cada dato recibido por el maestro este genera un ACK.

Los datos a transmitir son tres: corriente como dato1, temperatura como dato2 y voltaje como dato3. Luego de que el esclavo ya envió todos los datos y el maestro los contabilizó y verificó su validez, genera un NACK con lo que ya el esclavo no envía más datos y finaliza la transmisión. Por último el maestro genera la condición de parada con lo que culmina la transmisión con dicho esclavo.

Más adelante explicaremos los diagramas de tiempos de la transmisión con el valor que toman los bits de los registros de estado y control del módulo I2C.

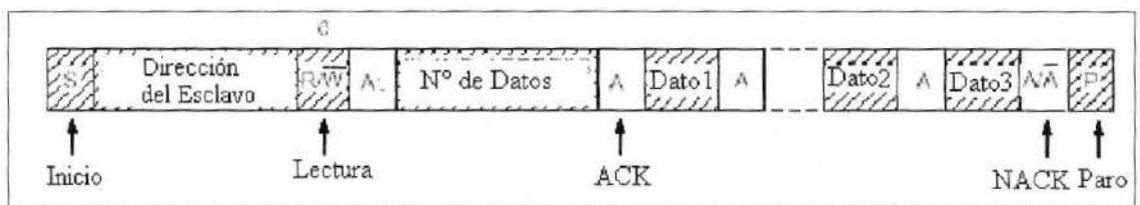


Figura 5.9.- Formato de la transmisión.



CIB-ESPOL

### *Protocolo de Comunicación*

En este sistema, un simple módulo maestro está en el bus y periódicamente iniciará comunicaciones con módulos esclavos. El protocolo de comunicaciones desarrollado permite que el módulo maestro lea de un esclavo en particular. La longitud de cada dato en este sistema es de 8 bits, en casos donde se use una longitud variable de datos debería ser especificado en la comunicación. En el sistema el requerimiento de lectura del módulo maestro empieza con una dirección específica, y la cantidad de datos que se van a leer del esclavo.

En el módulo maestro hay métodos que aseguran que el dato fue recibido exitosamente. El protocolo además maneja los errores de comunicación, identificando las fuentes de error como respuestas simultáneas a una misma dirección o no respuesta de un módulo esclavo en el bus.

### *Transmisión Maestro-Esclavo*

El módulo Maestro actúa como receptor y el módulo Esclavo actúa en el bus I2C como transmisor. Cada módulo esclavo espera ser direccionado por el módulo maestro para transmitir los datos que ha recogido de la línea de carga.

Existen un Firmware en el módulo maestro y otro en el módulo esclavo que manejan la comunicación entre los módulos.

#### *a) Firmware del módulo maestro*

Primeramente se realizan las configuraciones para la utilización del módulo I2C del Pic, se configura los pines de la Puerta C como entradas. La velocidad de transmisión la maneja el generador de Baudios, el que se carga con un valor que se encuentra en los 7 bits menos significativos del registro SSPADD, el generador de Baudios realiza conteos descendentes hasta cero cada vez que se cargue este valor en SSPADD. En el módulo I2C la recarga se hace automáticamente. La frecuencia de oscilación del Pic y el valor del registro SSPADD en la ecuación 5.1, nos da el valor de velocidad de la transmisión en Baudios de la comunicación.

$$\text{Baudios} = \text{FOSC} / (4 * (\text{SSPADD} + 1))$$

Ecuación 5.1.-

El Firmware de comunicación habilita el módulo I2C con el bit SSPEN (SSPCON<5>) del módulo maestro. Para iniciar la transmisión en modo receptor debe establecer el bit, RCEN (SSPCON2<3>) a uno. El generador de Baudios empieza su conteo y en cada desbordamiento, el estado del pin SCL cambia de alto a bajo y bajo a alto. Primero se genera una condición de inicio con el bit SEN (SSPCON2<0>) = 1, luego de generada esta condición el bit el bit SSPIF = 1 indicando que se ha generado la condición de inicio exitosamente. Seguido se envía la dirección del primer esclavo escribiéndola en el registro SSPBUF, así mismo se espera que el bit SSPIF = 1 indicando que se ha enviado la dirección, si el esclavo direccionado no responde con una condición de reconocimiento, ACKSTAT (SSPCON2<6>) es activado y se aborta la transmisión con el envío de una condición de Parada (STOP) y luego prosigue con la siguiente dirección, pero si el esclavo responde se procede a recibir el dato. El primer dato es el número de datos que va a transmitir ese esclavo y luego envía los datos especificados en el primero, el dato ingresa por la línea

SDA al registro de recepción SSPSR (el cual no se puede acceder vía software). Después del octavo pulso de reloj el bit RCEN automáticamente se pone a cero, luego el contenido de SSPSR se carga en el registro SSPBUF, el bit BF = 1 (indicador de Buffer lleno), el bit SSPIF = 1 que indica de que hay un dato recibido en SSPBUF y el generador de Baudios es inhabilitado de contar, manteniendo la línea SCL en bajo. Cuando el buffer es leído, el bit BF es automáticamente puesto a cero, mientras que el Bit SSPIF se pone a cero vía software. Cuando el Firmware maestro recibe el dato, pasa al control de memoria donde se organizan los datos pertenecientes a cada esclavo y se guardan en las memorias RAM y EEPROM del Pic. El módulo I2C del Pic se pone en estado IDLE (Espera) hasta la espera del siguiente comando, que es el envío de una condición de reconocimiento ACK, terminando su estado IDLE e indicándole al esclavo que el dato ha sido recibido con éxito y que está listo para recibir el siguiente. Mientras se reciben los datos del esclavo, el Firmware del maestro realiza un conteo de datos recibidos, comparándolos con el número de datos enviados por el módulo esclavo, cuando la recepción de datos del esclavo se ha completado, se envía un NACK, ACKDT = 1 (SSPCON2<5>) y ACKEN = 1 (SSPCON2<4>) para iniciar la secuencia de reconocimiento, indicando que se quiere finalizar la transmisión con ese módulo esclavo. Finalmente una condición de parada termina la transmisión con el módulo

esclavo. Luego se envían los datos de ese esclavo a la PC con el módulo USART que se explica más adelante luego de lo cual sigue la transmisión con el siguiente esclavo. El Firmware del maestro va direccionando uno a uno esclavos hasta la capacidad especificado en el módulo I2C del Pic. Una vez se han direccionados todos, se procede a enviar las direcciones de todos los esclavos que respondieron y el número total de esclavos a la PC para su análisis en el software de monitoreo.

#### *Errores en la Comunicación*

El módulo I2C tiene indicadores de errores en la comunicación, por lo que el Firmware del maestro se encarga de monitorearlos y evitarlos. El bit BF se activa cuando se ha recibido un dato en SSPBUF del registro SSPSR y se desactiva cuando se lee el registro SSPBUF. El bit SSPOV se activa cuando se ha recibido un dato en SSPSR y el bit BF está activado pero de una recepción anterior, es decir cuando el registro SSPBUF no ha sido leído. El bit WCOL se activa cuando se inicia una escritura en SSPBUF cuando una recepción está en progreso, como resultado el contenido de SSPBUF no cambia. En la figura 5.10 se puede observar el diagrama de tiempos de los bits del Firmware del maestro y en la figura 5.11 su diagrama de flujos.

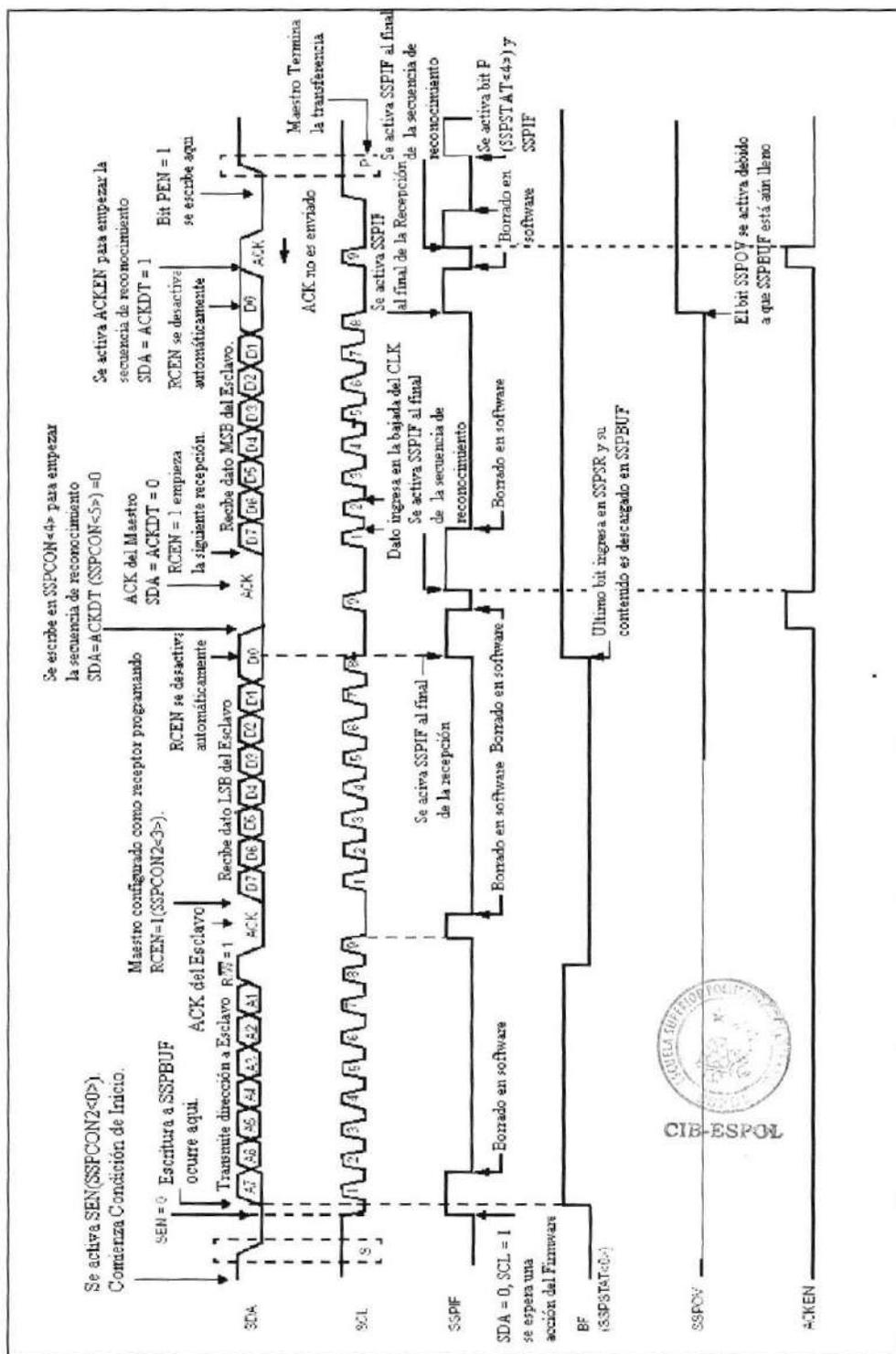
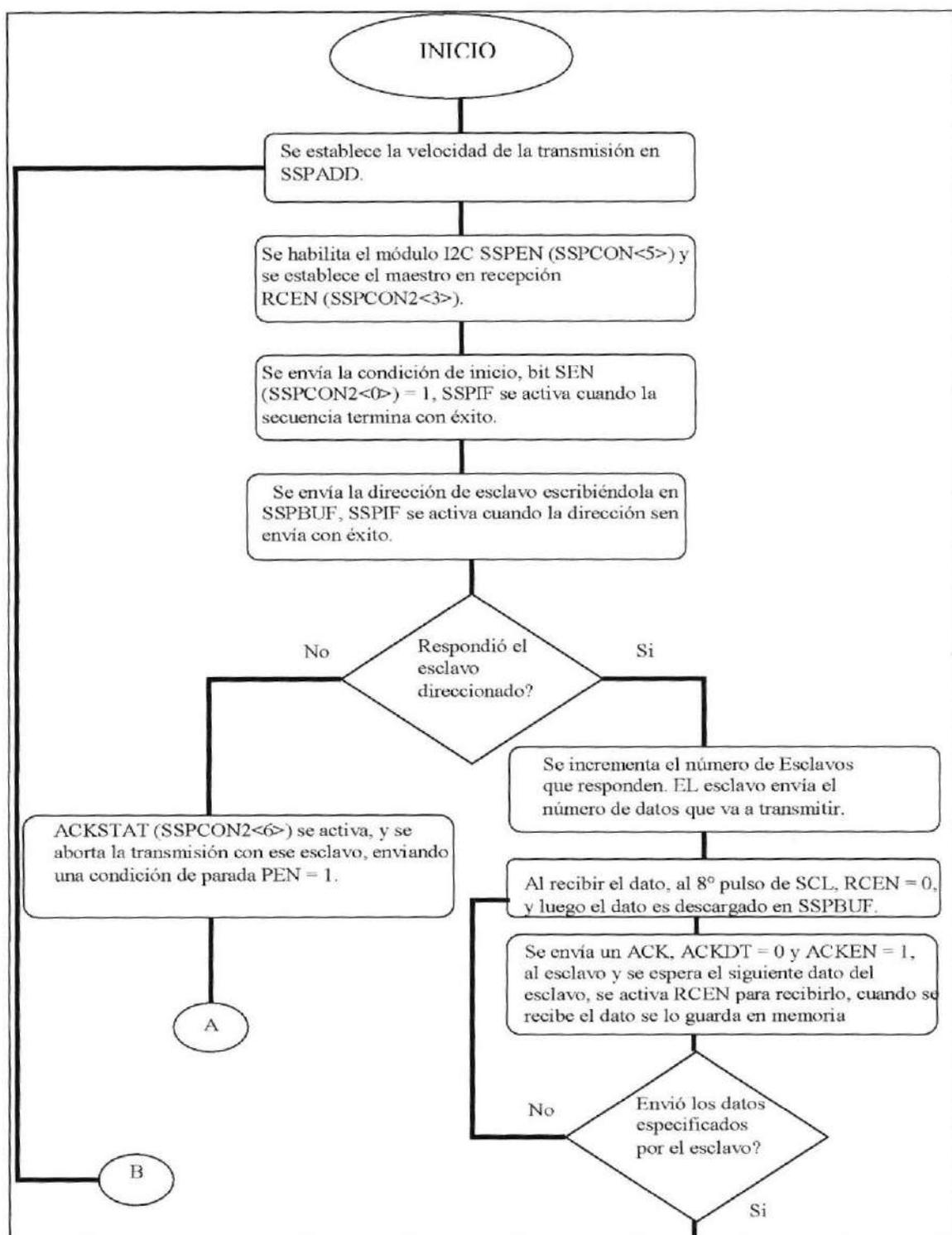


Figura 5.10.-Diagrama de Tiempos del Maestro.





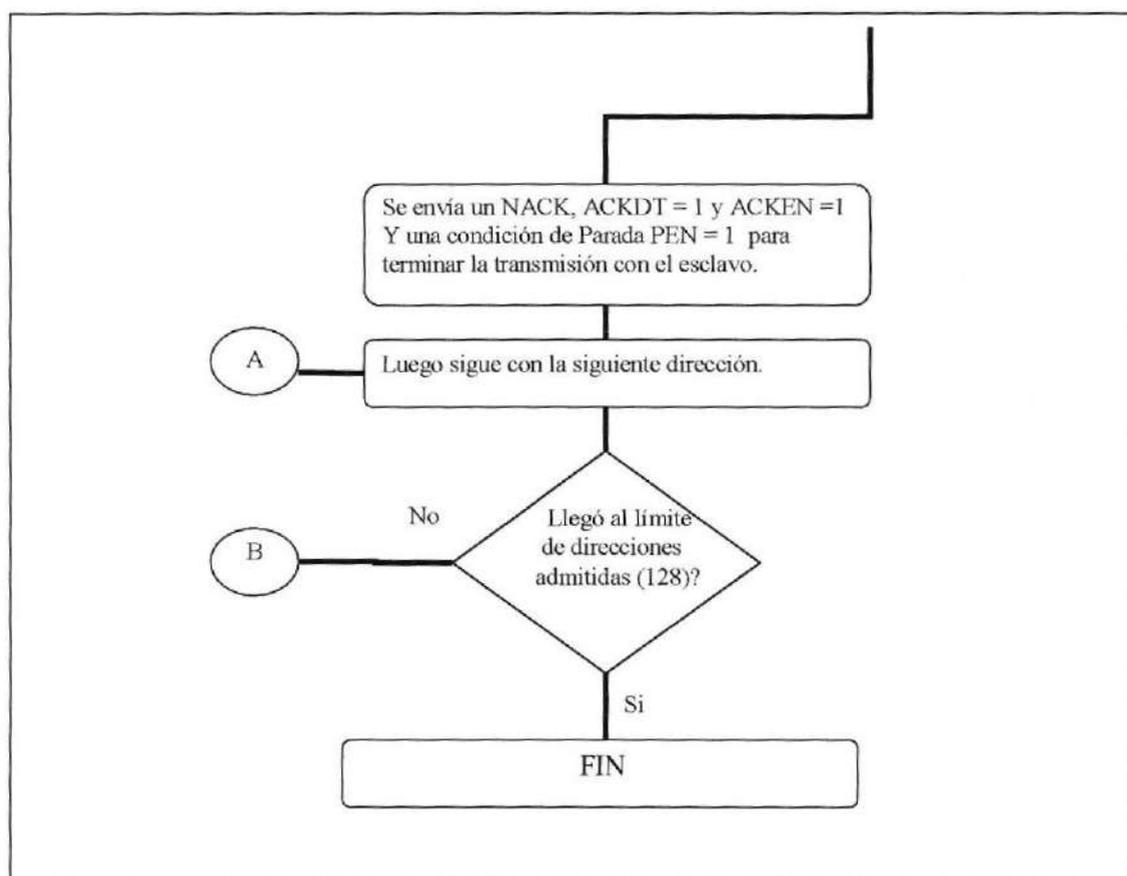


Figura 5.11.- Diagrama de flujos Firmware Maestro.

#### b) Firmware del módulo Esclavo

El Firmware esclavo habilita el módulo I2C con el bit SSPEN (SSPCON<5>) y lo configura en modo esclavo con dirección de 7 bits con los bits SSPM3:SSPM0.

Con el bit CKP (SSPCON<4>) se habilita el reloj que interviene en la comunicación. En el registro SSPADD se guarda la dirección que asignamos al esclavo, cada módulo esclavo tiene una dirección diferente. Luego se habilita la interrupción por eventos del módulo I2C activando el bit SSPIE (PIE1<3>), para que se produzca la interrupción cuando concuerde la dirección que envía el maestro con la del esclavo.

El Firmware esclavo luego de configurar el módulo I2C, entra en un estado llamado Sleep o de bajo consumo propio del Pic, a la espera de que el direccionamiento del maestro coincida con la propia del Pic, en cuyo caso se produce la interrupción, con lo cual sale del estado de bajo consumo. Cuando coincide la dirección y el bit R/W del byte de dirección entrante está activado, el bit R/W del registro SSPSTAT se activa indicando que el módulo maestro va a receptor los datos del esclavo. La dirección recibida va al registro SSPBUF. El pulso del ACK es enviado automáticamente en el noveno bit, y el pin SCL se mantiene en bajo. Luego de salir del estado de bajo consumo se realizan los procedimientos de recolectar los datos de la línea de carga y luego de procesarlos, el dato que va a ser transmitido se carga en SSPBUF, el cual

además se carga en SSPSR automáticamente y salen bit a bit por la línea SDA. Luego, el pin SCL se habilita con el bit CKP (SSPCON<4>). El maestro monitorea el pin SCL para enviar otro pulso de reloj mientras que el esclavo mantiene al maestro sin enviar pulsos manteniendo la línea SCL en bajo con CKP = 0 pero con CKP = 1 el maestro empieza a mandar pulsos de reloj. Los 8 bits del dato salen en la bajada de los pulsos de SCL, esto asegura que la señal en SDA sea válida durante el tiempo en alto de la línea SCL. El indicador SSPIF se activa en la bajada del noveno pulso de reloj y se desactiva por software.

Luego de enviar cada dato, el maestro responde con la línea SDA en bajo (ACK). Como se trata de un esclavo transmisor, el pulso ACK del maestro receptor se recoge en la subida del noveno pulso de reloj en SCL. Luego el pin SCL es habilitado activando el bit CKP, para enviar el siguiente dato.

Cuando se han enviado todos los datos del módulo esclavo y la línea SDA está en alto (NACK), entonces la transferencia de datos está completa. Cuando el NACK es recogido por el esclavo, la lógica del esclavo produce un reset y el

esclavo entra de nuevo en estado de bajo consumo y monitorea que el direccionamiento del maestro concuerde con su dirección.

### *Errores en la comunicación*

El Firmware esclavo también revisa los bits que indican que hubo algún error. Se lee siempre el registro SSPBUF para que el bit BF se desactive. El bit SSPOV indica que llegó un dato mientras existe ya uno en SSPBUF por lo tanto hay que desactivar este bit para que el esclavo al ser diseccionado pueda generar un ACK al maestro. Esta condición es llamada desbordamiento, y se define cuando cualquiera el Bit BF o el bit SSPOV es activado. En la Tabla XXVII se observa el estado de estos bits y las acciones que se producen.



CIB-ESPOL

TABLA XXVII

Acciones de una transferencia de datos de acuerdo a los bits BF y SSPOV.

Estado de los bits en una recepción		SSPSR → SSPBUF	Genera Pulso ACK	Activa el bit SSPIF
BF	SSPOV			
0	0	Si	Si	Si
1	0	No	No	Si
1	1	No	No	Si
0	1	Si	No	Si

En la figura 5.12 se observa el diagrama de tiempos de los bits que intervienen en la transmisión de datos con el maestro y en la figura 5.13 el diagrama de flujos del Firmware del Esclavo.

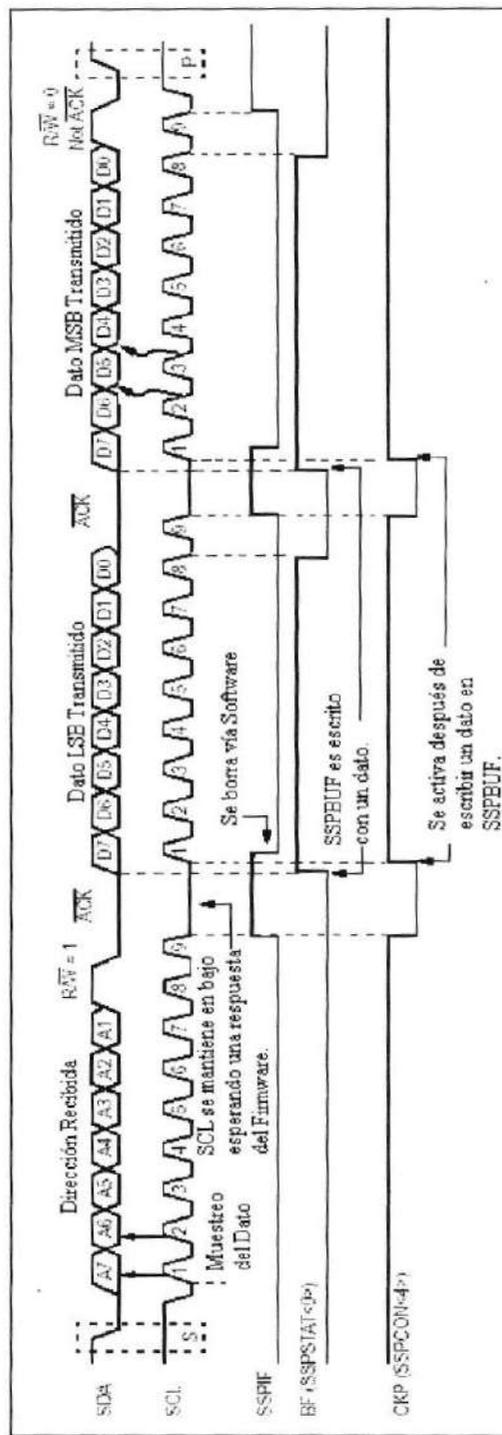
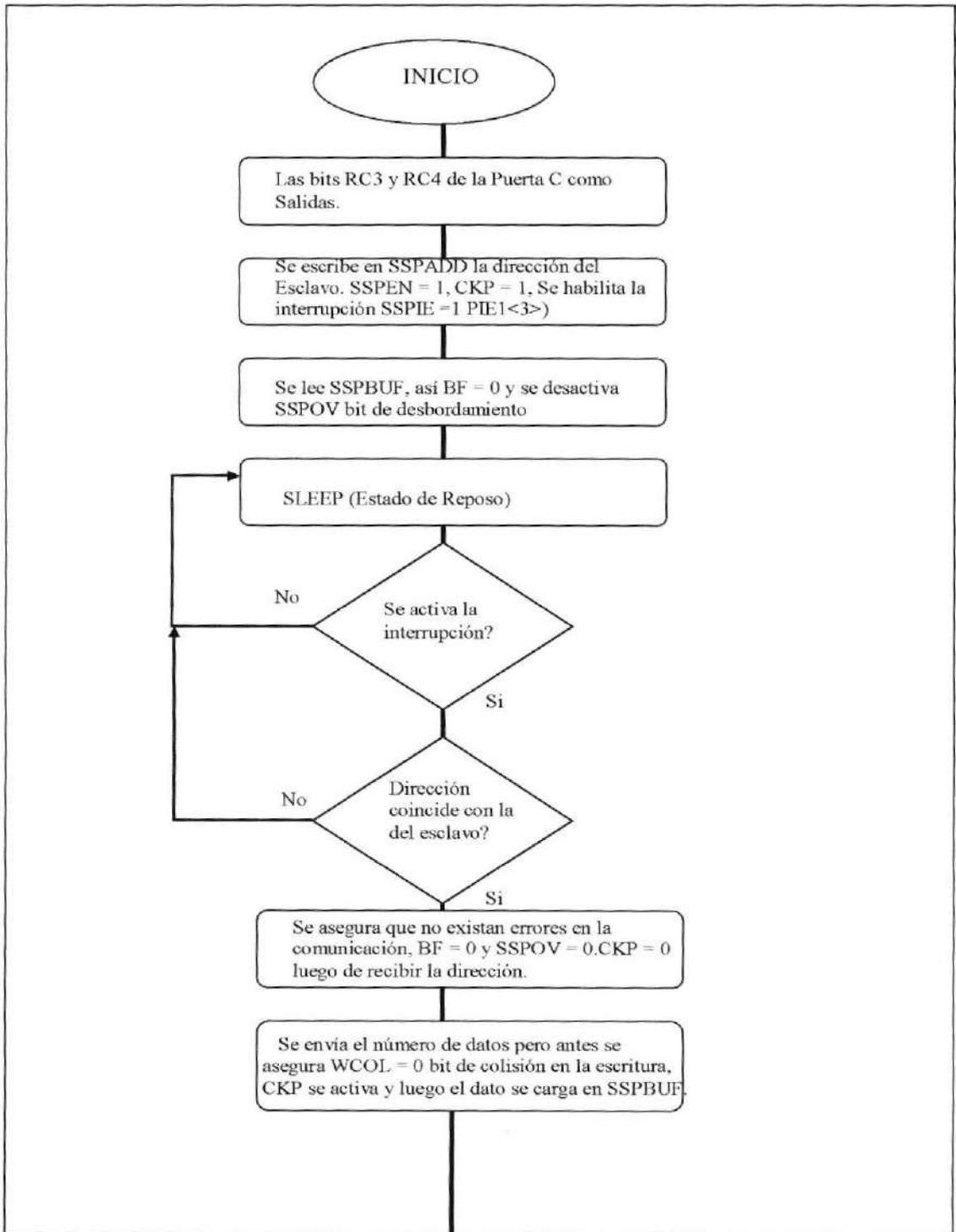


Figura 5.12.- Diagrama de Tiempos del módulo Esclavo.



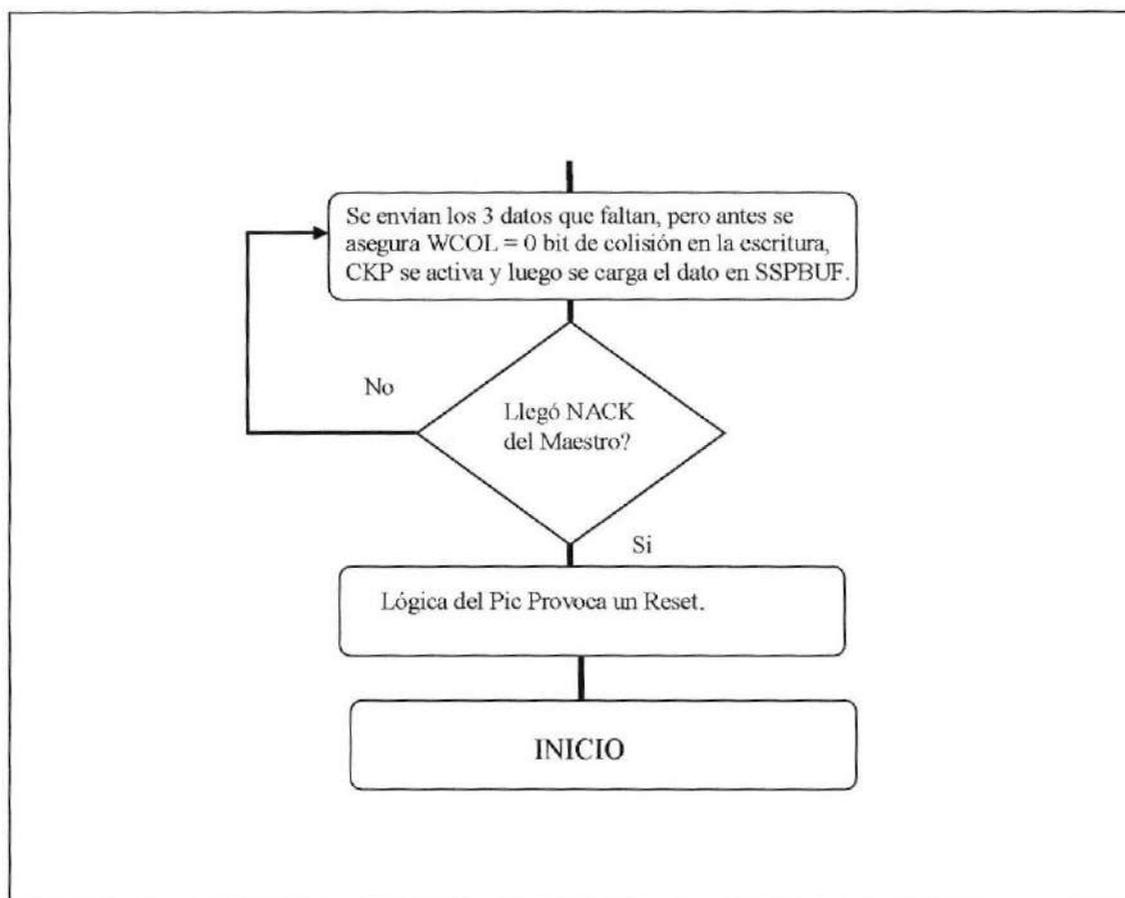


Figura 5.13.- Diagrama de Flujos Firmware del Esclavo.

## 5.2.2 MODULO DE COMUNICACIÓN USART.

El módulo de comunicación USART, se usa en el módulo maestro para enviar los datos recolectados de los módulos esclavos a la PC usando el protocolo

RS-232. El módulo USART usa dos modos de comunicación: Síncrona y Asíncrona, pero en este proyecto se usa la más común, la Asíncrona. En el modo asíncrono no hay señal de reloj por lo tanto un pin es usado para transmisión y el otro para recepción de datos. Esta forma de operación es conocida como "Full Duplex".

La transmisión y recepción pueden ocurrir al mismo tiempo pero para el proyecto primero se usa la recepción y luego se realiza la transmisión de los datos.

#### *Formato de la transmission*

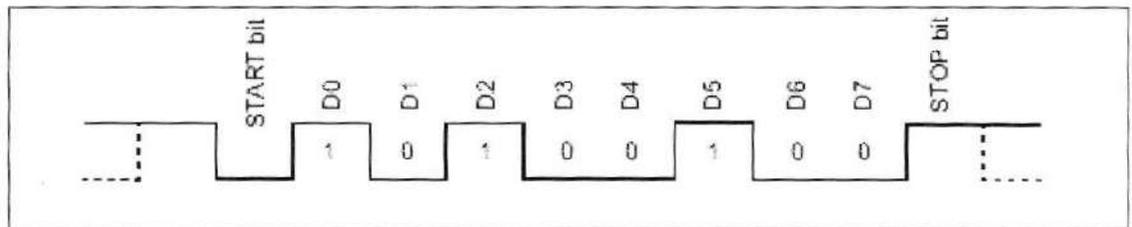


Figura 5.14.- Señal en forma binaria '10100100' en modo Asíncrono.

Las señales en los pines TX y RX están en alto cuando ninguna transmisión o recepción está en progreso y pasa a un bajo cuando la transmisión empieza. Esta transición en bajo es usada por el receptor para sincronizar el dato entrante. La señal se mantiene en bajo por la duración del bit START y es seguido por los bits del dato, el bit menos significativo primero. El último bit de un byte es seguido por un bit de STOP el cual es un alto. La transmisión por consiguiente termina con el pin en alto. Después que la salida del bit STOP se ha completada, ocurre la salida del bit START de la siguiente transmisión.

Existen varias cosas que notar en la señal de la Figura 5.14, la cual representa la señal en los pines TX o RX del microcontrolador. El bit START es un cero y el bit STOP es un uno. El dato es enviado con el bit menos significativo primero así que el patrón del número se ve al revés en comparación a la manera en la que aparece cuando se escribe un número binario. El dato no es invertido aun cuando el protocolo RS-232 usa voltajes negativos para representar un uno lógico.

Generalmente, cuando se usa el USART para las comunicaciones RS-232, la señal debe ser invertida a través de un chip de esta clase (MAXIM-RS232A).

El módulo maestro envía los datos de la siguiente forma (Figura 5.15):

- Son 3 datos por esclavo por lo que ( $3*N$ ) cantidad de datos esclavos precedidos de un bit START y seguidos de un bit de STOP.
- Número de datos ( $n^\circ$ ) por esclavo y dirección de cada esclavo.
- Número total de módulos esclavos **N**.

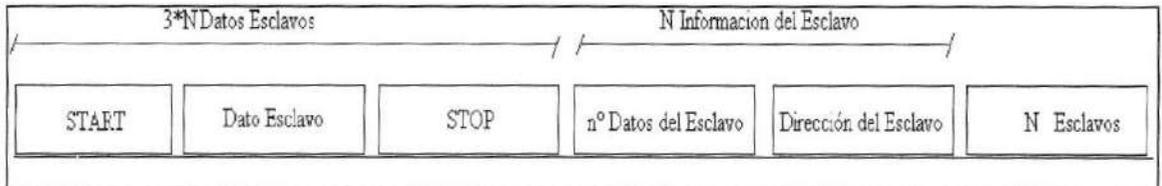


Figura 5.15.- Formato del envío de datos recolectados por el módulo maestro a la PC.

### *Transmisión módulo Maestro-PC*

El Firmware del maestro maneja y habilita el módulo USART para la transmisión de datos, donde la transmisión se realiza por la línea TX (RC6) y la recepción por la línea RX (RC7) por lo que se configuran los pines RC7 como entrada y RC6 como salida de la Puerta C. Como ya se vio en el capítulo 3, los registros de estado y control de la comunicación son TXSTA y RCSTA mientras que los registros TXREG y RCREG son para la escritura o lectura de los datos respectivamente. Mientras que el registro SPBRG establece la velocidad de la comunicación. Los datos que se transmiten son de 8 bits, un bit de START y un STOP. Luego de configurar los pines de la Puerta C, se habilita el módulo USART activando el bit SPEN (RCSTA<7>) y luego como Receptor activando el bit CREN (RCSTA<4>) de recepción continua. Prosigue estableciendo la velocidad de la comunicación en el registro SPBRG siendo esta de 9600 BAUD para nuestra aplicación, además el bit BRGH (TXSTA<2>) = 1 en modo alta velocidad y en modo asíncrono SYNC (TXSTA<4>) = 0. El software del Sistema de Monitoreo empieza la recolección de los datos enviando una petición de datos al módulo maestro, el Firmware está atento a la llamada y verifica que sea una llamada correcta. Cuando se recibe una llamada, esta ingresa por RX un bit a la vez al registro serial de desplazamiento RSR, luego de que el último bit

ha ingresado y el bit de STOP es identificado, el dato pasa a un buffer y luego a RCREG solo si este se encuentra vacío. El buffer y RCREG forman un tipo de pila lo que le da tiempo al Firmware de extraer el dato de la pila antes de que se produzca un error de desbordamiento (Buffer y RCEG). Es posible que se hayan recibido dos bytes y estar ocupado recibiendo un tercer byte antes de que el dato en RCEG sea leído, entonces se produce un error de desbordamiento.

Luego de recibir la llamada correcta del sistema, el módulo maestro deshabilita el módulo USART en modo recepción y habilita el módulo I2C para recolectar datos de un módulo esclavo. Cada vez que ha recibido datos de un esclavo deshabilita el I2C y se conecta al módulo USART para transmitir a la PC, luego se desconecta del USART y vuelve al I2C para recoger datos del siguiente esclavo, así continua hasta que se recoja los datos de todos los esclavos. Todos los datos van al módulo de control de memoria y se guardan para la transmisión a la PC, el Firmware maestro deshabilita el módulo I2C y habilita el USART SPEN (RCSTA<7>) en modo transmisor TXEN (TXSTA<5>), habilitado el USART se inicia la secuencia de transmisión donde se guarda el número de datos del esclavo y su dirección en la EEPROM de datos, seguido se extraen los datos del esclavo del módulo de control de memoria, cada dato se carga en

el registro TXREG para enviar el dato. El bit TXIF (PIR<4>) se activa cuando el byte en el registro TXREG pasa al registro serial de desplazamiento TSR. De TSR salen uno a uno los bits por el pin TX precedidos por un bit de START y seguido de un bit de STOP. El uso de un registro TSR permite que un nuevo dato sea escrito en TXREG mientras un previo dato aún está siendo transmitido, con esto se logra un máximo rendimiento en el envío de los datos. Cuando se envían todos los bits del byte por TX se activa TMRT (TXSTA<1>) indicando que el dato se ha transmitido y luego continua con el siguiente.

En el proceso de recolección de datos de todos los esclavos, el Firmware maestro guarda el número de datos y dirección de cada esclavo en la EEPROM de datos, los organiza y luego contabiliza el número de esclavos que respondieron a la llamada del sistema. Cuando el maestro extrae todos los datos esclavos del control de memoria (RAM) y los envía, el Firmware lee los datos de información de cada esclavo de la EEPROM y el número total de esclavos y los envía a la PC con el módulo USART. En las figuras 5.16 y 5.17 se observa los diagramas de tiempos de los diferentes bits que intervienen en el envío de datos con el módulo USART a la PC.



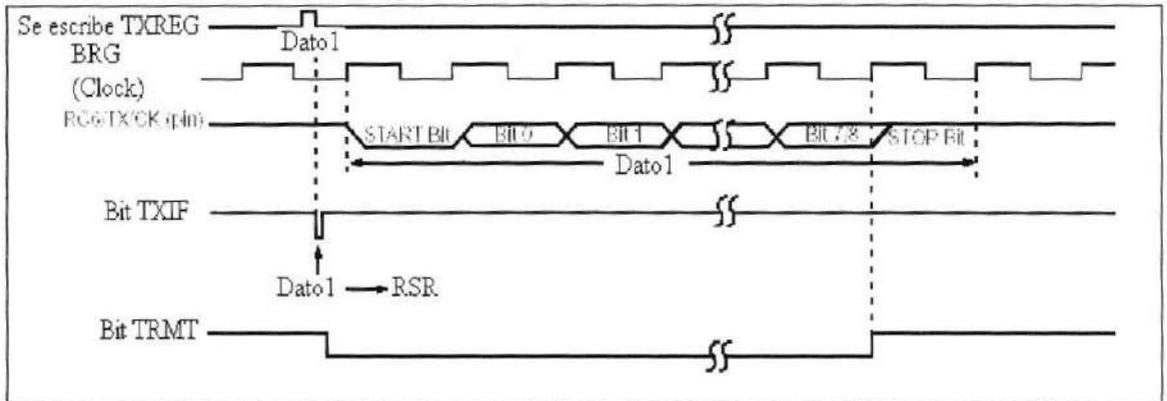


Figura 5.16.- Diagrama de tiempos de la transmisión de un byte con el módulo USART.

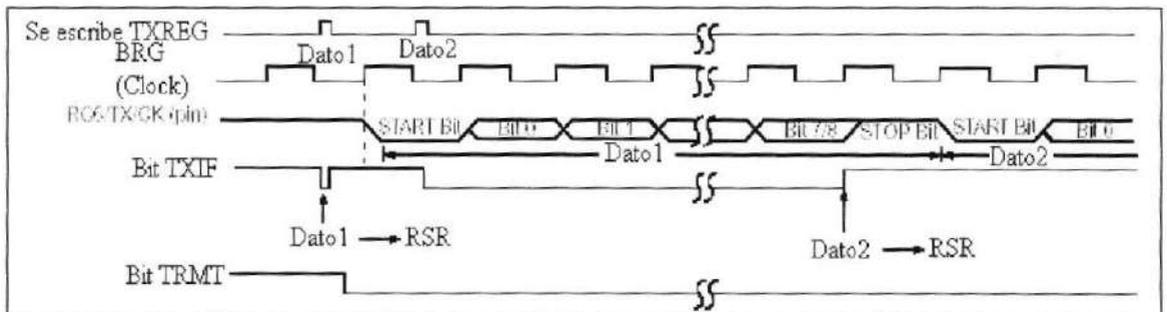
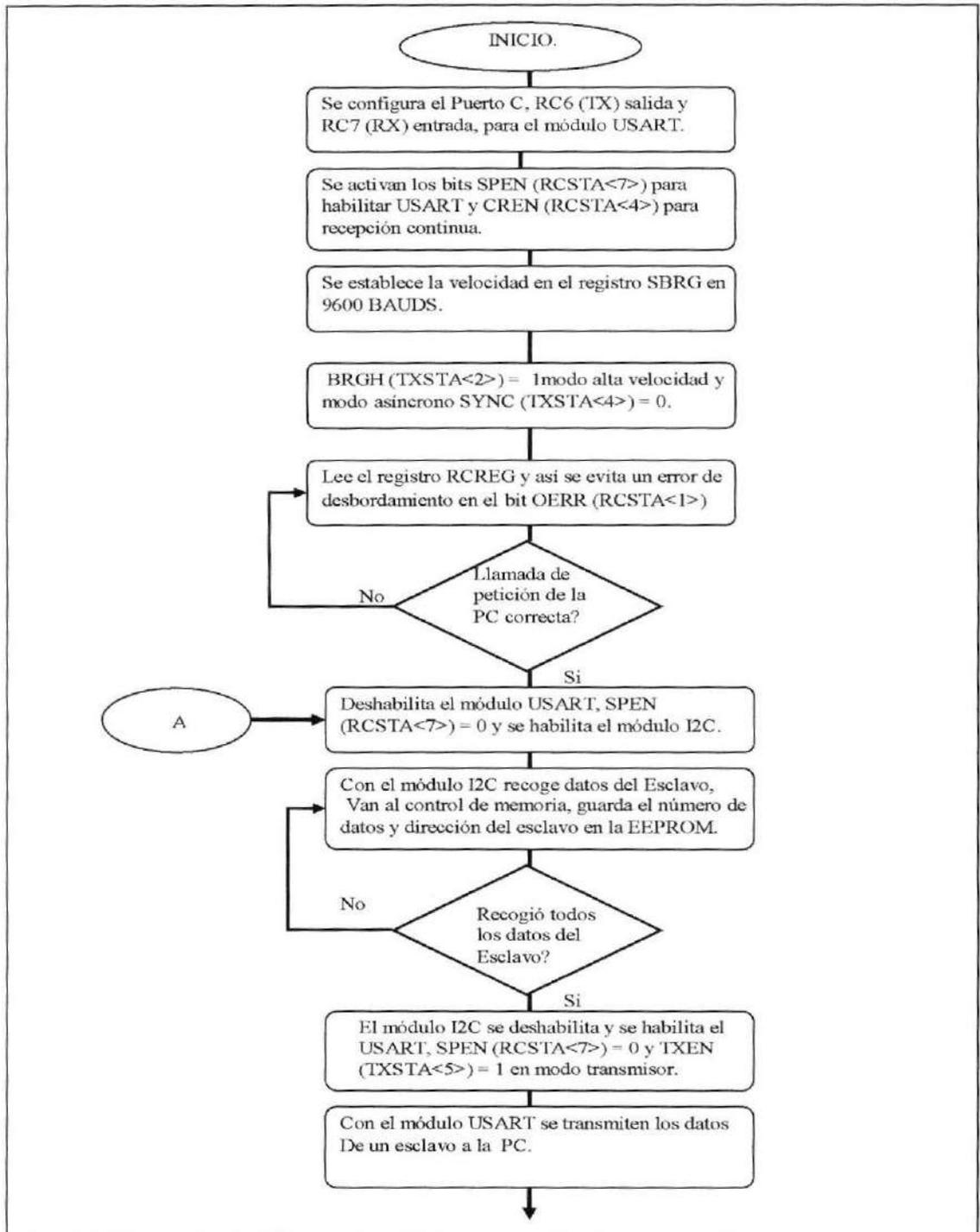


Figura 5.17.- Diagrama de tiempos de la transmisión de más de un byte con el módulo USART.

El diagrama de flujo del módulo de comunicación USART se encuentra en la figura 5.18:



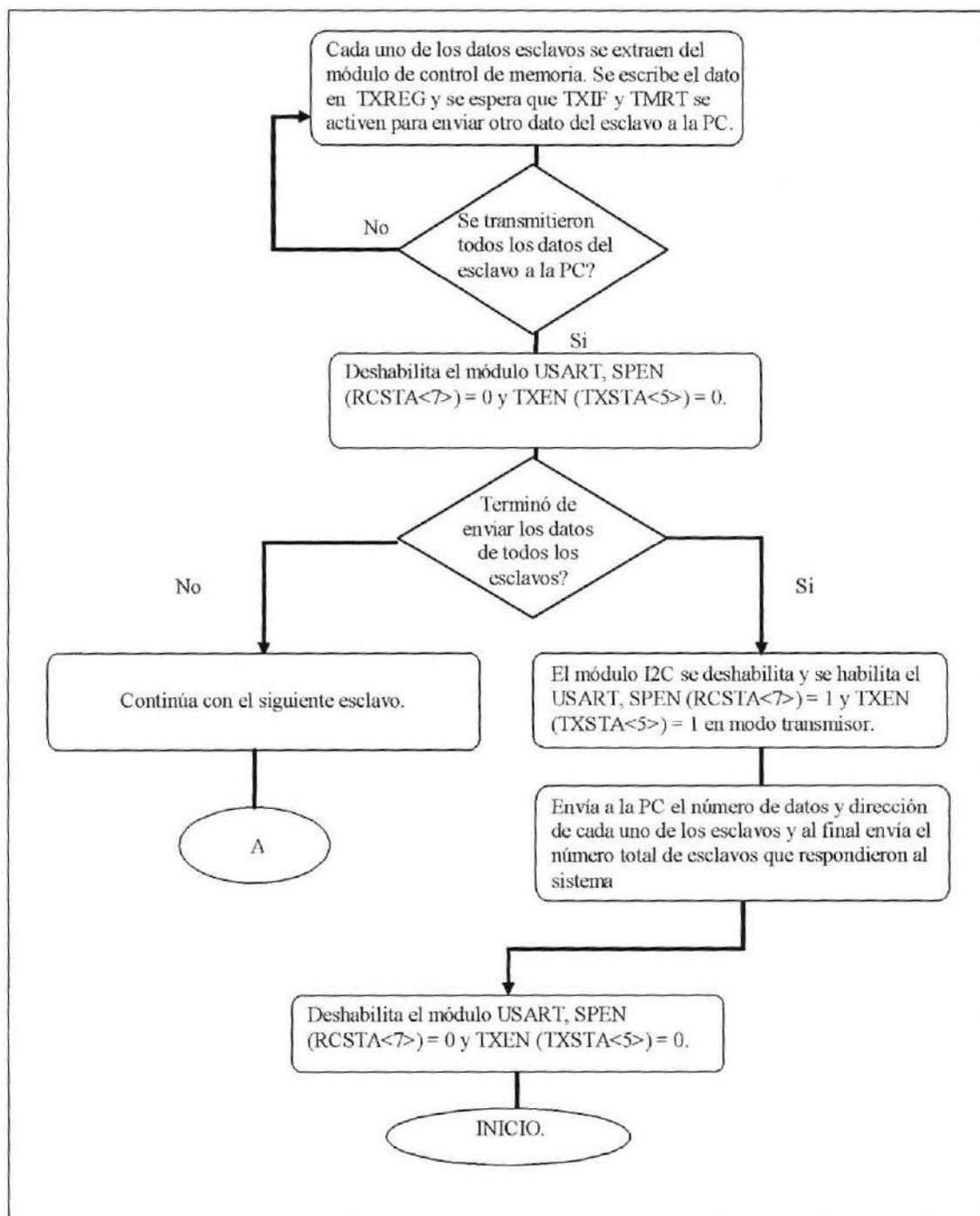


Figura 5.18.- Diagrama de Flujo del módulo de comunicación USART.

### *Errores en la comunicación*

Existen dos tipos de errores en la comunicación: error de trama y error de desbordamiento. El error de la trama se puede verificar con el bit FERR (RCSTA<2>), este error ocurre cuando el bit de STOP es cero. El bit de STOP siempre debería ser uno. El error de trama es siempre asociado con el byte en RCREG y pasa a través de la pila de la misma manera que el dato con el que es asociado. Leyendo RCREG permite que el siguiente byte sea cargado en RCREG con su propio bit de error de trama (FERR). Por esta razón es esencial leer el bit FERR antes de que el dato sea leído de RCREG. No es necesario limpiar el bit FERR, será actualizado tan pronto como un nuevo byte de datos sea recibido en RCREG.

El error de desbordamiento se puede establecer con el bit OERR (RCSTA<1>) y ocurre cuando pila está llena con dos bytes que ya han sido recibidos y un tercer byte está llegando al registro RSR. Dado que este tercer byte necesita ser

trasladado dentro de la pila pero al no haber espacio disponible, este byte es descartado y un error de desbordamiento se produce.

Los errores de desbordamiento pueden ser evitados leyendo el dato entrante de RCEG lo suficientemente rápido. Una vez que ocurre un error de desbordamiento, ningún dato nuevo será recibido hasta que la lógica de reopción haya sido re-iniciado, desactivando el bit CREN y habilitándolo de nuevo. Un síntoma de esto es que el USART deja de recibir datos inesperadamente, a menudo después de los dos primeros bytes.

#### *Interfase con el RS-232*

El Maxim RS-232 como se ve en la figura 5.19, es usado para convertir los niveles lógicos del microcontrolador a los niveles usados por el protocolo RS-232, donde por lo general, hay una señal de tierra (GND) y dos señales: una salida de transmisión (TXD) y una entrada de recepción (RXD).



### 5.2.3 MODULO DE CONTROL DE MEMORIA EEPROM

El módulo de control de Memoria, es parte del Firmware maestro y administra los datos recolectados por cada esclavo en la memoria RAM y EEPROM. Como se ve en la figura 5.20 el microcontrolador tiene una memoria RAM con cuatro bancos donde se encuentran los diferentes registros de control y registros de propósito general, en este último se guardan temporalmente los datos de cada esclavo. Cada banco tiene 96 bytes para almacenar información excepto el banco 2 que tiene 80 bytes.

Para acceder a las diferentes posiciones de la memoria en los 4 bancos el Firmware usa un direccionamiento indirecto. Los registros utilizados son: FSR, IPR y INDF. El registro INDF no es un registro implementado físicamente. Cualquier instrucción que use el registro INDF accederá al registro al que apunta el registro FSR, es decir en FSR se encuentra la dirección de la localidad de memoria y en INDF el dato esclavo. Escribiendo una dirección en FSR y escribiendo en INDF guardaremos un dato en la RAM pero si leemos INDF obtendremos un dato de la RAM. Para movernos a través de los 4 bancos de memoria usamos el registro IRP conjuntamente con FSR.

El Firmware maestro direcciona un esclavo y guarda el número de datos que envía el esclavo en la localidad 0024H del banco 0, luego cada dato del esclavo se guardan en las direcciones siguientes pero antes pasa por el módulo de control de memoria para verificar que la localidad de memoria a la que se quiere acceder no sobrepasa el límite de localidades del banco actual (007FH) , caso contrario se escoge el siguiente banco con IRP y FSR, se guarda el dato en INDF e indirectamente se guardará en el siguiente banco. Cada dato que llega del esclavo pasa por el módulo de control de memoria, determina la localidad de memoria y el banco adecuado donde se guarda el dato. Una vez se termina la comunicación con ese esclavo se obtiene el número de datos del esclavo de la RAM, la dirección del esclavo, se lo guarda en la EEPROM de datos y continúa con la siguiente localidad de memoria, a continuación pasa al módulo de control de memoria para extraer los datos esclavos y mandarlos a la PC con el módulo USART y continúa la comunicación vía módulo I2C con el siguiente esclavo y se repite lo anterior. Cuando se termina la comunicación con todos los esclavos y el envío de sus datos a la PC, el Firmware procede a leer la EEPROM de datos y extraer el número de datos por esclavo y su dirección y enviarla a la PC.



Luego de enviar toda la información de cada esclavo, se envía el número total de módulos esclavos que respondieron a la llamada del sistema.

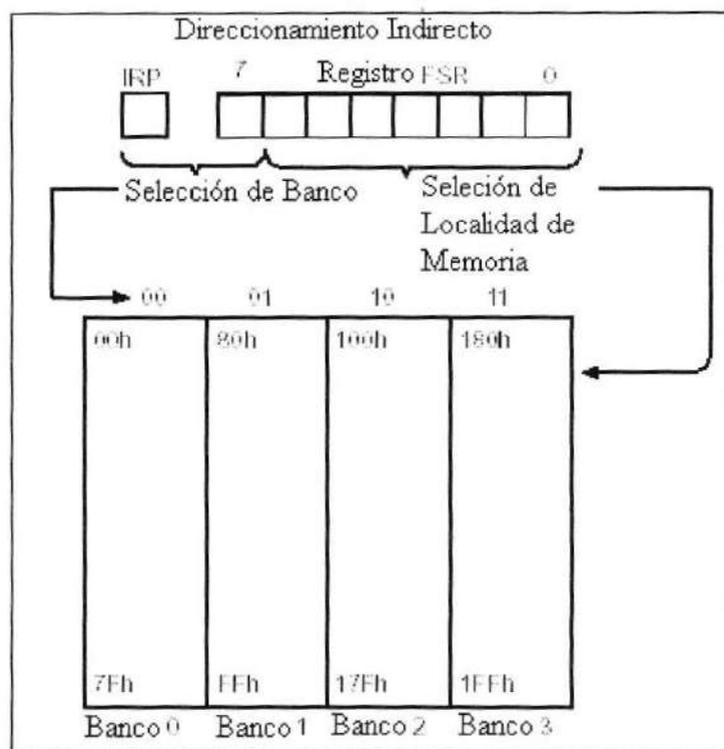
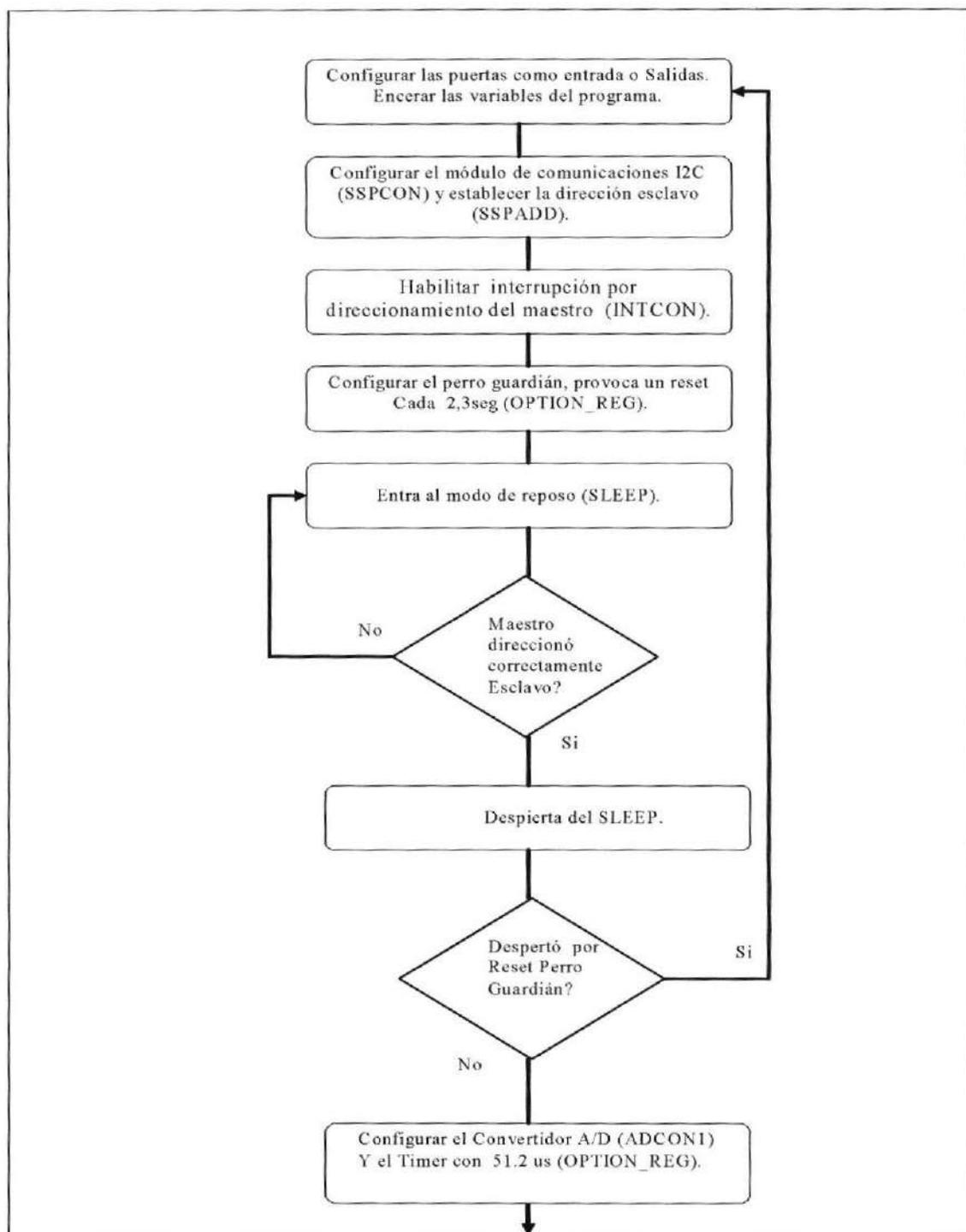


Figura 5.20.- Direccionamiento indirecto utilizado por el módulo de control de memoria.

### **5.3 MODULO ESCLAVO.**

En este sistema hay un solo módulo maestro y dos módulos esclavos, los módulos esclavos se pueden conectar en cualquier momento sin necesidad de parar el sistema, el software detecta automáticamente la introducción de un esclavo al sistema y además se puede conectar sin afectar el funcionamiento de los demás, siempre que no exceda el número máximo de esclavos que puede soportar el sistema. A continuación en la figura 5.21 podemos ver el diagrama de bloques del Firmware del esclavo:



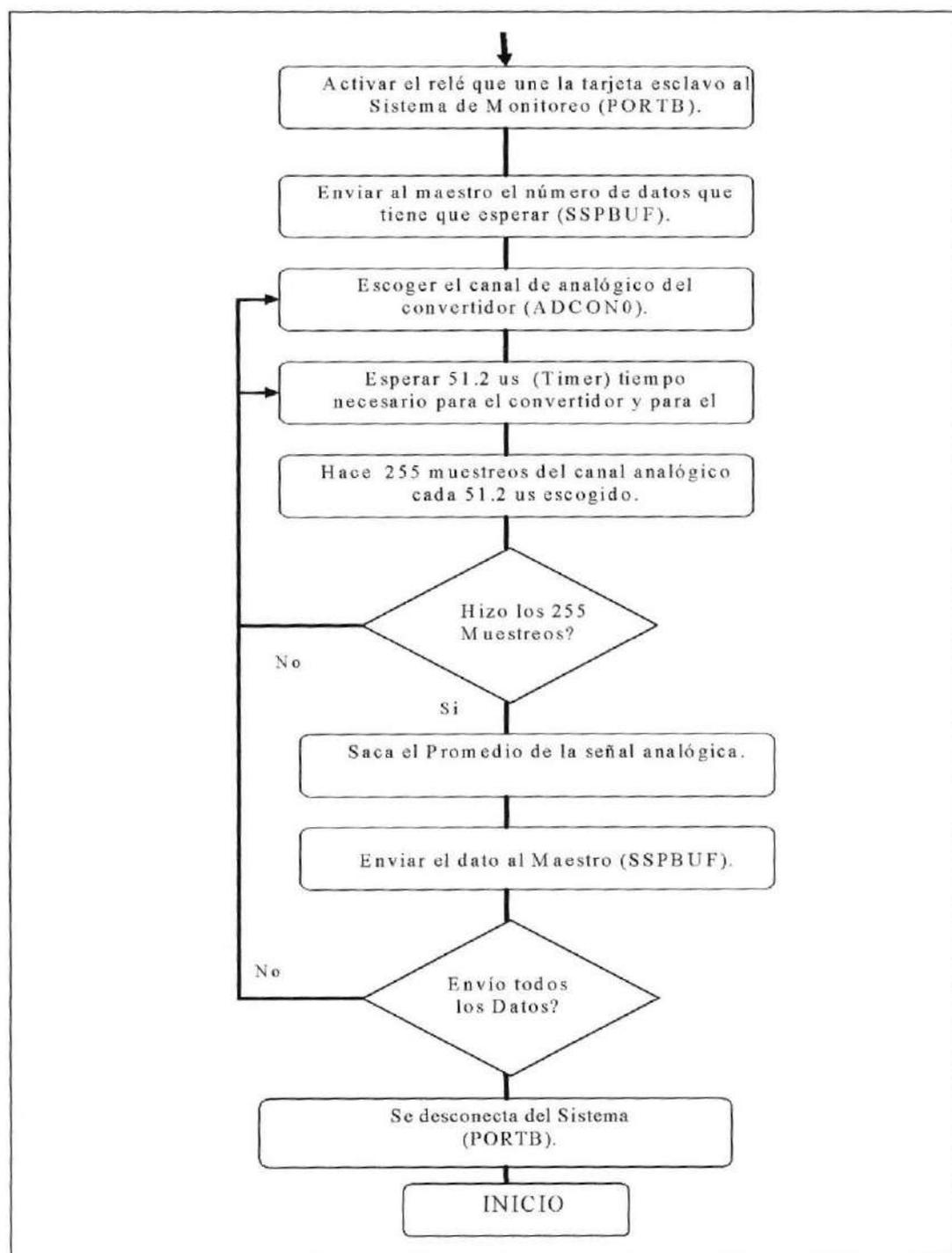


Figura 5.21.- Diagrama de Flujos del funcionamiento del Módulo Esclavo.

En la figura 5.22 se puede apreciar al diagrama de bloques de un módulo Esclavo. Las señales del sistema que son corriente, temperatura y voltaje, llegan al bloque de amplificación y acondicionamiento de señales, en la cual las señales se filtran en el caso de la corriente y se la amplifica con una ganancia de  $A = 31$ , en el caso del voltaje se disminuye su voltaje con un divisor de voltaje y en la temperatura que es una señal de 0-20 mA, se le hace el tratamiento adecuado para tener un voltaje proporcional a la lectura, para que el Pic pueda convertirlo a una señal digital. Luego una vez que se tratan las señales, entran al microcontrolador al bloque Convertidor A/D, el cual convierte las señales en un número digital con una resolución de 10 bits. Luego sigue al bloque de muestreo de señales, en el que se muestrea la señal analógica durante un tiempo pero si la señal tiene ruido o no es simétrica para seguridad de que el programa no se quede colgado en este bloque del Firmware tiene la ayuda del bloque WDT o perro guardián, que está atento, si se quedó la ejecución del programa en alguno de los bloques provocará un reset después de 2,3 segundos de espera, pero sino luego pasa al bloque de promedio de la señal, con lo que obtenemos el valor promedio de la señal muestreada y finalmente pasa al módulo I2C que es una comunicación serial de 3 hilos

bidireccional desarrollada para comunicación entre IC o circuitos integrados, sirve de comunicación Esclavo-Maestro. Adicionalmente el Control de conexión al sistema, activa un relé que une la tarjeta esclavo al sistema, uniendo la referencia analógica del esclavo con la referencia digital del sistema con lo que tenemos la misma referencia y se pueden tomar lecturas correctas, todo esto en el momento en que el Maestro ha direccionado correctamente al esclavo.

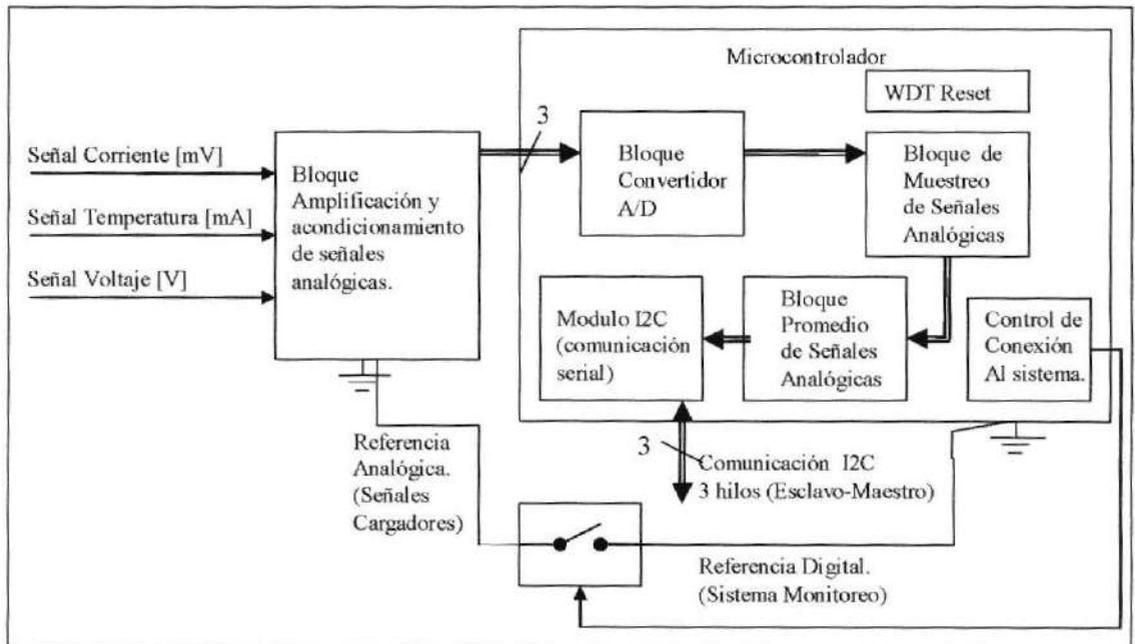


Figura 5.22.- Diagrama de bloques del módulo Esclavo.

### 5.3.1 MODULO DE AMPLIFICACION Y ACONDICINAMIENTO DE SEÑALES

Las variables del sistema son 3 como ya se mencionó. Estas señales se introducen en una electrónica sencilla de amplificación y acondicionamiento como se ve en la figura 5.23 para obtener un voltaje proporcional a la lectura, en un rango de 0 a 5 V, niveles necesarios para la conversión a un número digital de 10 bits.

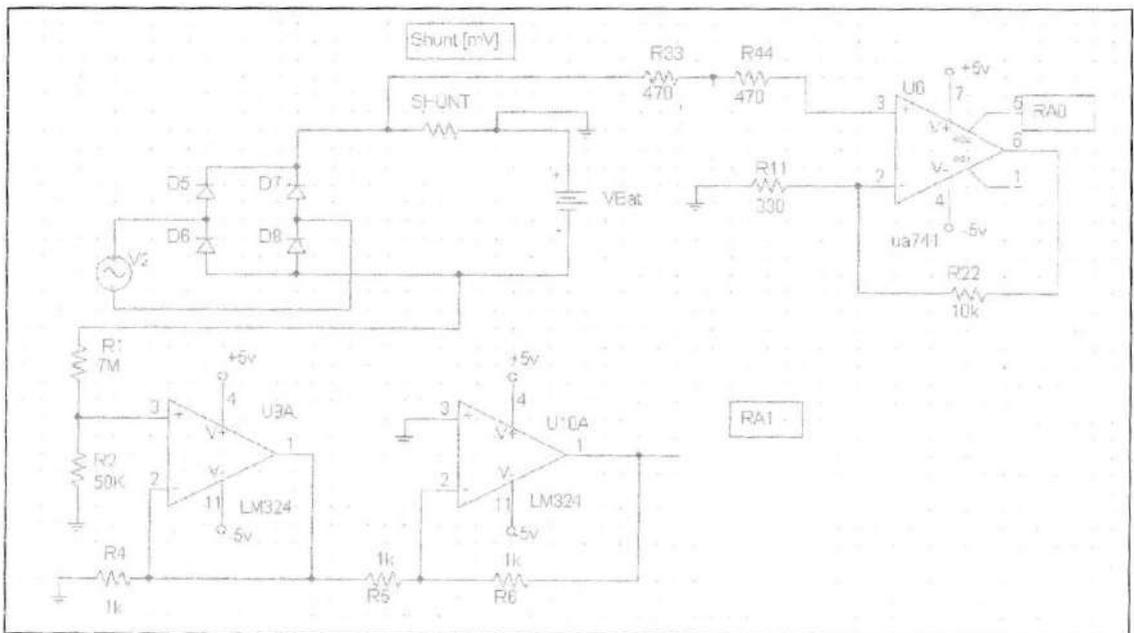


Figura 5.23.- Figura del circuito de monitoreo de un circuito de baterías.

### *Corriente*

El sistema monitorea la corriente que circula por la línea de carga de baterías, en esta sección explicaremos el tipo de corriente que carga a las baterías, la electrónica empleada por los módulos esclavos para el tratamiento de la señal de la corriente.

La corriente es de tipo DC pero dependiendo del circuito del cargador tendrá un rizado, en unos cargadores será más pronunciado que en otros. En la figura 5.24 se encuentra graficado la corriente en un cargador. En la gráfica A tenemos pulsos de corriente que pueden ser controlados cambiando dos diodos del puente por dos SCR, y variando sus pulsos en el circuito, dependiendo del circuito de fuerza del cargador. En la gráfica B está graficado en violeta el voltaje AC de entrada que luego es rectificado y en verde la corriente que va a la batería, se puede apreciar que circula en las zonas donde el voltaje AC está por arriba del voltaje de la batería. Los cargadores con control de corriente pueden controlar el ancho de los pulsos de corriente controlando los pulsos de disparo a los SCR.



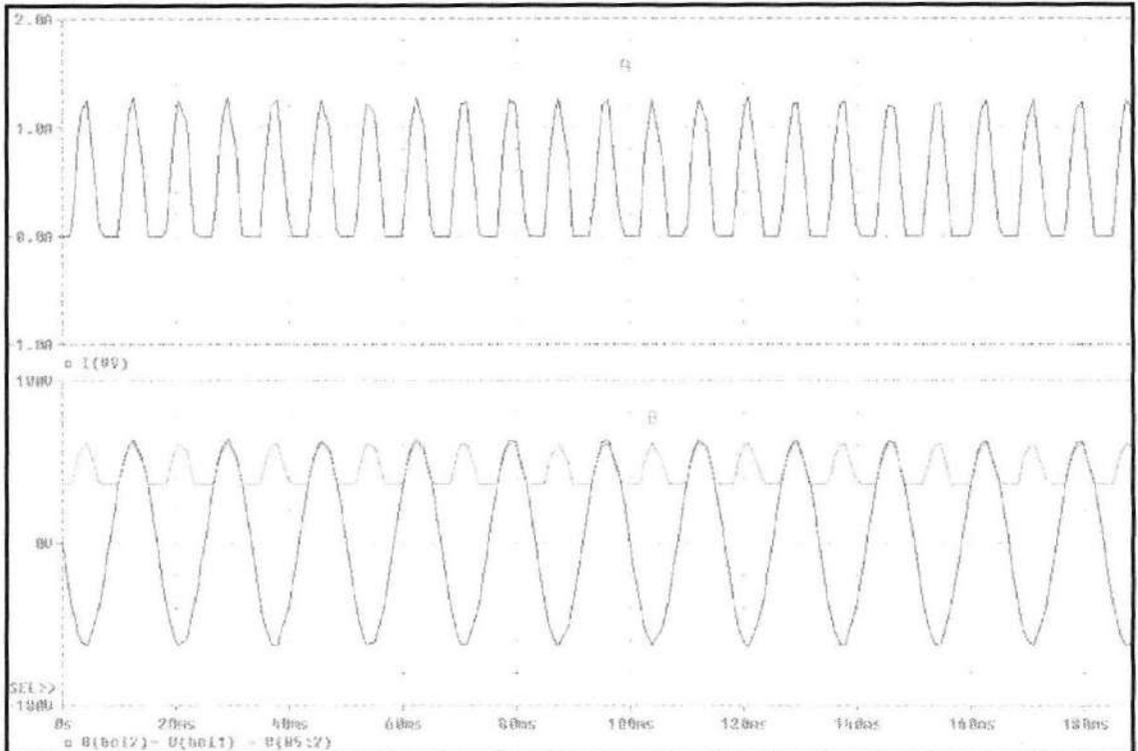


Figura 5.24.- A) Forma de onda de la corriente que circula por las baterías.

B) formas de ondas del Voltaje de entrada AC y el tipo de corriente que recibe la batería.

En ciertos cargadores se pone una inductancia en el circuito para alisar la corriente y tener una corriente DC pura (Figura 5.25), dependiendo de la inductancia cambiará el porcentaje de rizado de la corriente.

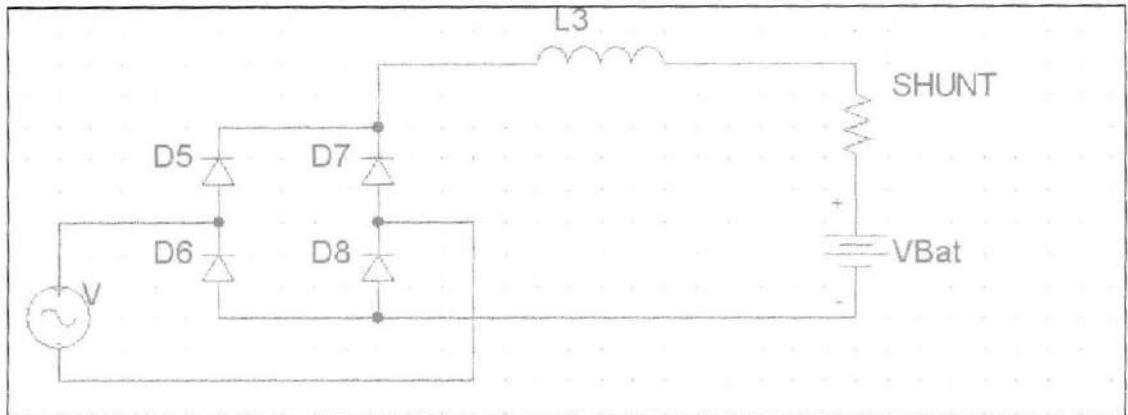


Figura 5.25.- Circuito de fuerza de un Cargador de Baterías con inductancia, la batería recibe una corriente más DC pura.

En la figura 5.25 y 5.26 podemos observar el circuito de fuerza de un cargador de baterías con y sin inductancia, pero en los dos observamos el shunt.

El shunt no es nada más que una barra de cobre con cierto espesor que depende de la corriente que circula por él y que tiene una resistencia muy baja por lo tanto si su resistencia es baja y circula corriente en el orden de Amperios, la caída de voltaje en el shunt es en milivoltios. Esta señal es proporcional a la corriente que circula por el shunt, teniendo la misma forma de onda, por consiguiente su rizado.

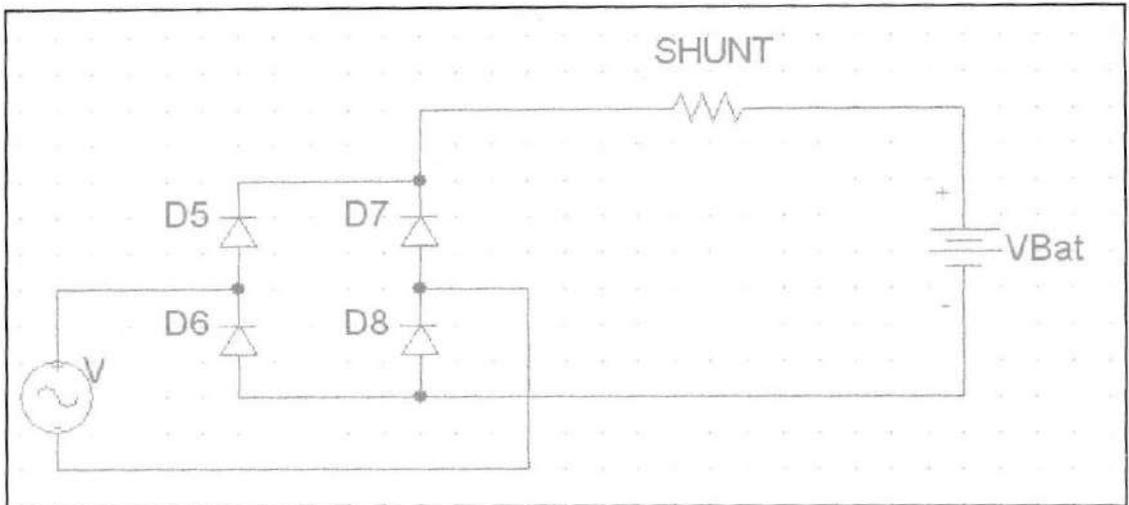


Figura 5.26.- Circuito sin inductancia, la batería recibe pulsos de Corriente.

Para que el módulo esclavo realice el monitoreo, debe tener una medida de la corriente que está pasando por el cargador, esta señal es la que nos brinda el Shunt, como ya mencionamos es en milivoltios, por lo tanto la tarjeta esclavo dispone de la electrónica de la Figura 5.27 necesaria para censarla y tratar la señal para su posterior lectura.

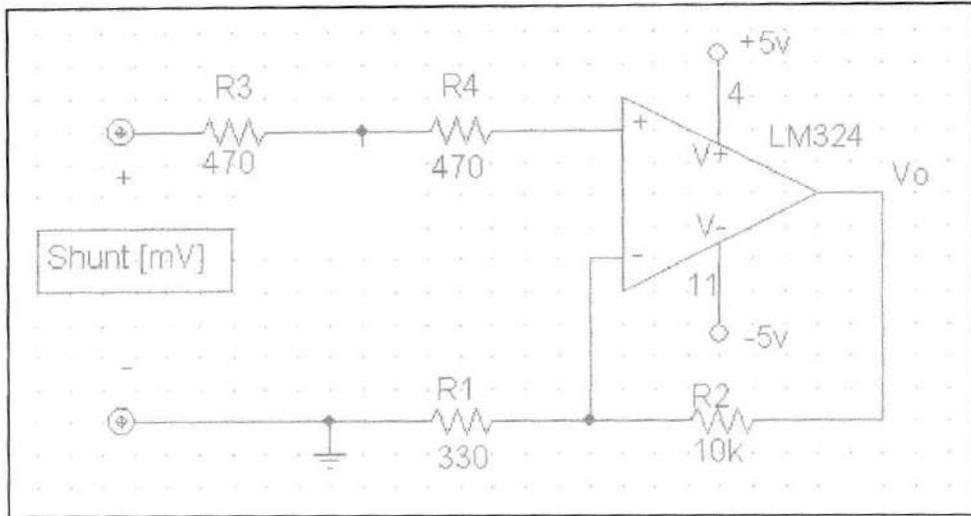


Figura 5.27.- Circuito de tratamiento y amplificación de la señal de la Corriente.

El circuito es una configuración no inversora, en el pin positivo del operacional se encuentra un pequeño configuración, que estabiliza la señal del shunt, luego se amplifica con una ganancia de 31, según la ecuación 5.2 de amplificación de un OPAMP no inversor. La salida va al Pic para su procesamiento.

$$V_o = Shunt * \left(1 + \frac{R_2}{R_1}\right); \quad A = \left(1 + \frac{R_2}{R_1}\right)$$

Ecuación 5.2.-

### *Voltaje*

El voltaje de la batería es otro parámetro importante del banco, con él podemos determinar si el banco de Baterías está cargado, así como el porcentaje de carga del banco.

El voltaje de la batería en carga sube un poco con respecto a su estado en vacío. Si las baterías suben rápidamente en carga es porque inicialmente ya estaban con un alto porcentaje de carga por el contrario si el monitoreo se hizo en cargadores para formación de baterías y estas están con un 0% de carga su voltaje demorará en subir.

En el capítulo 3 se vio que en el software de monitoreo a un 60% de carga divide el voltaje total de la línea para el voltaje de cada batería al 100% teórico de carga, que aproximadamente es de 15 V en carga, debido a que cada celda debe tener un voltaje de 2.5 Voltios para que dicha celda esté cargada, cada batería de 12 Voltios tiene 6 celdas por lo que el voltaje que debe alcanzar la batería es de  $15V ( 2.5*6 )$ , por lo tanto el software hace un cálculo aproximado

de cuantas baterías hay en la línea de baterías, con ese dato luego en la pantalla de estadístico del software hace el cálculo de baterías cargadas al mes. En la Figura 5.28 se encuentra el circuito de un cargador de baterías y el voltaje de la línea de baterías que forma parte del monitoreo.

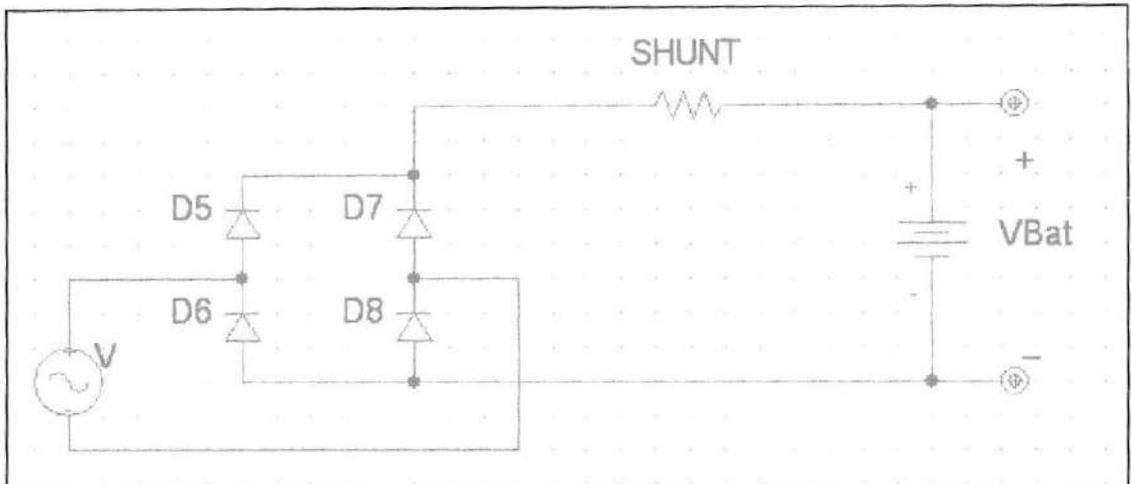


Figura 5.28.- Circuito de Fuerza de un cargador, donde VBat es la señal que va a la tarjeta Esclava para su lectura.

A medida que se realiza la carga de la línea o banco de baterías el voltaje va subiendo y el tiempo que demora en cargar la línea depende mucho del tipo de carga que se realiza, como ya se explicó en el capítulo 3, cargas rápidas con corrientes altas, el voltaje subirá mucho más rápido.

En la Figura 5.29 se encuentra la circuitería para el tratamiento de la señal de voltaje en la línea de baterías que se está monitoreando, cabe indicar que el voltaje que ingresa a la tarjeta esclava en una línea de batería es de más de 200 [V] DC, dependiendo del cargador de baterías, por lo que el divisor de voltaje debe tener una resistencia grande como se ve en la figura 5.28 aproximadamente de 7 M, para que la alta impedancia de este ramal que va en paralelo con el circuito de Fuerza no permita pasar corrientes grandes. El primer operacional es de ganancia unitaria y el segundo tiene una configuración inversora, ya que como podemos observar el voltaje que va a la tarjeta es negativo debido a que tomamos como referencia analógica la línea positiva de acuerdo a la posición del shunt y línea de baterías en el circuito de fuerza, por lo que hay que invertirlo así el microcontrolador puede leerlo. El divisor de voltaje reduce el voltaje de la línea (según la ecuación 5.3, Señal de voltaje de la línea de baterías). Llevada a un nivel entre 0-5V para su lectura., donde el límite superior corresponde al voltaje máximo que se puede medir en la tarjeta.

$$V_0 = \left( -\frac{R_6}{R_5} \right) \times \left( \frac{R_2}{R_1 + R_2} \right) \times -V_{Bat}$$

Ecuación 5.3.-

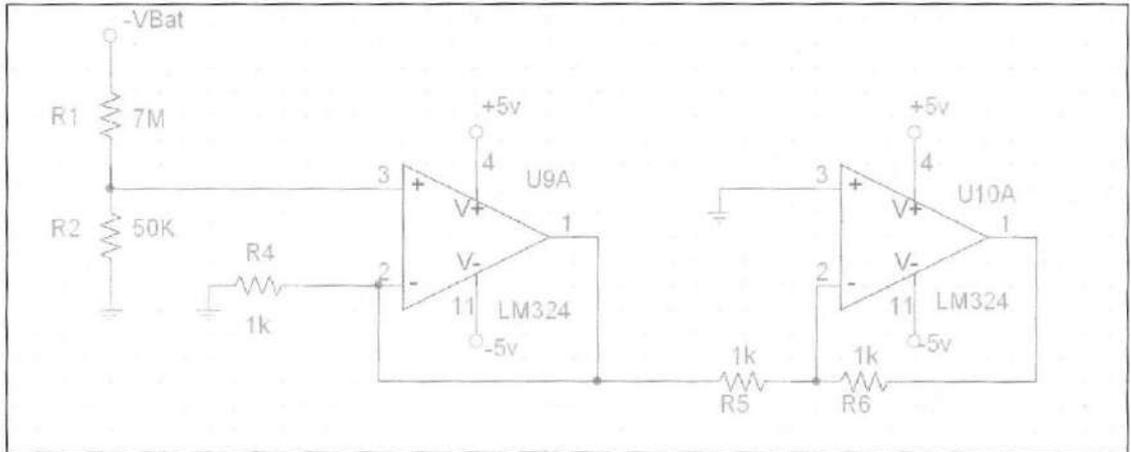


Figura 5.29.- Circuitería de tratamiento de la señal de Voltaje de la línea de Baterías.

### Temperatura

El tercer parámetro es la temperatura y se monitorea debido al proceso de carga de baterías en el que circula corriente a través de ellas, donde se produce una reacción química y se desprenden gases, por consiguiente hay un incremento en la temperatura. Los componentes internos pueden exceder los límites de temperatura permisible que pueden causar daños internos en la batería como por ejemplo las placas de cada celda.

Por lo general, en la práctica se controla que la temperatura no sobrepase de 60° C. El software de monitoreo, que se explica en el capítulo 3, alerta de este evento en la pantalla de alarmas que nos muestra la línea de baterías que tiene alta temperatura.

Para monitorear la temperatura se usa el IC LM334Z usado como sensor de temperatura (Sus características se pueden ver en el Capítulo 4), y además una tarjeta transductora que convierte la señal de corriente del LM334Z en niveles de corriente de 0 – 20 mA que ingresan a la tarjeta esclavo para ser transformadas en niveles de 0-5 voltios para luego la posterior lectura y procesamiento.

#### 1.- Sensor LM334Z



El lugar de medición (Bancos de baterías) y el de procesamiento de la lectura (Cuarto de Cargadores) están separados por una distancia considerable debido a la presencia de gases corrosivos que afecta los equipos de carga y lectura, por lo que el IC LM334Z es una solución adecuada frente a otros IC, ya que es

una fuente de corriente que no se ve afectada por distancias considerables y por su dependencia a la temperatura ya que envía una señal de corriente proporcional a la temperatura en grados kelvin a través del cable, usándose dos hilos nada más. Un hilo es para la alimentación del LM334Z de 1 a 40V y el otro hilo para la señal de corriente. El rango de la señal de corriente depende de la configuración del circuito sobretodo RSET.

La resistencia RSET es conectada entre R y V-, como se ve en la figura 5.30., su valor afecta el resultado de la corriente ISET que circula por el LM334Z. Es recomendable conectar RSET muy cerca del IC donde se monta el sensor para evitar pérdidas en los cables que afectarán el funcionamiento adecuado del sensor.

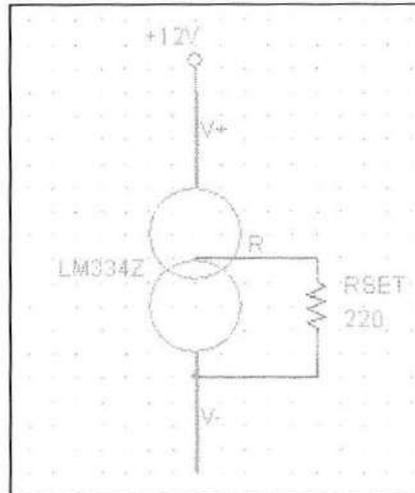


Figura 5.30.- Conexiones del LM334Z.

Según la ecuación 5.4, (Fórmula de dependencia del LM334Z con la temperatura), que se deduce en el capítulo 4,  $I_{SET}$  es inversamente proporcional a  $R_{SET}$ , en este proyecto se tomó el valor de  $R_{SET}$  para que se simplifique el factor  $227 \mu V/ ^\circ K$ , así tomando un valor de  $R_{SET}$  igual a  $220 \Omega$  tenemos un factor aproximado a  $1 \mu A/ ^\circ K$ .

$$I_{SET} = \frac{(227 \mu V/ ^\circ K) (T)}{R_{SET}}$$

Ecuación 5.4.-

La temperatura máxima que mide el sensor es de 70°C (343°K). Debido a la linealidad del IC LM334Z, podemos fácilmente determinar el valor de la temperatura (° K) a partir de la señal del sensor (mA), el factor exacto de la ecuación tiene una tasa de incremento de 1,032  $\mu$ A por cada ° Kelvin, por lo tanto teóricamente ISET a 70 °C sería de 0,354 mA. Esta tasa o factor de incremento puede ajustarse, variando RSET mediante un trimer según el figura 5.31. Se ve el ajuste de la tasa de incremento de ISET que es inversamente proporcional al valor de RSET. Se puede ajustar la salida de corriente correspondiente a la ecuación 5.4 usando un trimer como RSET.

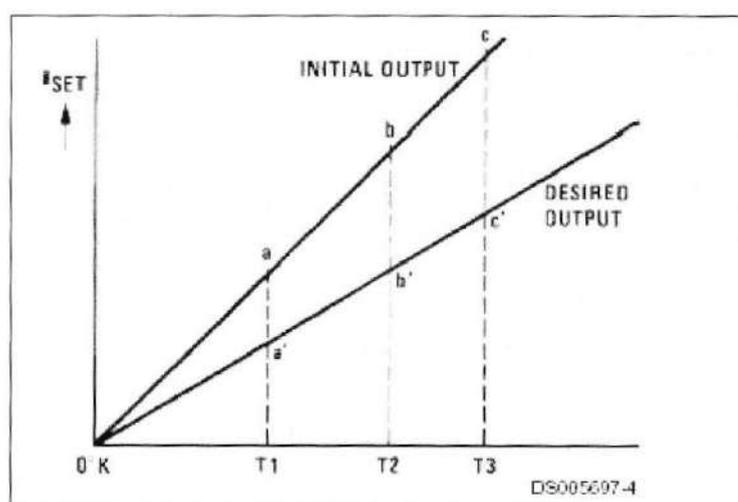


Figura 5.31.- Ajuste de la tasa de incremento de ISET, la pendiente de cada gráfico es inversamente proporcional a RSET.

La línea abc es la salida inicial que tenemos, al hacer un ajuste con el trimmer tenemos la línea a'b'c', la señal deseada, y observamos tanto que a como a' corresponden a la misma temperatura T1 pero tienen diferentes ISET.

## *2.- Tarjeta transductora de 0-20 mA.*

La tarjeta convierte la señal del IC LM334Z (ISET) en una señal de 0 – 20 mA, para conseguir la linealidad de los cálculos y facilitar su procesamiento, pasa por una electrónica, que amplifica, y ajusta a la salida indicada.

### *Ajuste y Calibración*

Esta tarjeta tiene un proceso de ajuste y calibración, para lograr la salida deseada de acuerdo a la entrada ISET, que se enumera a continuación:

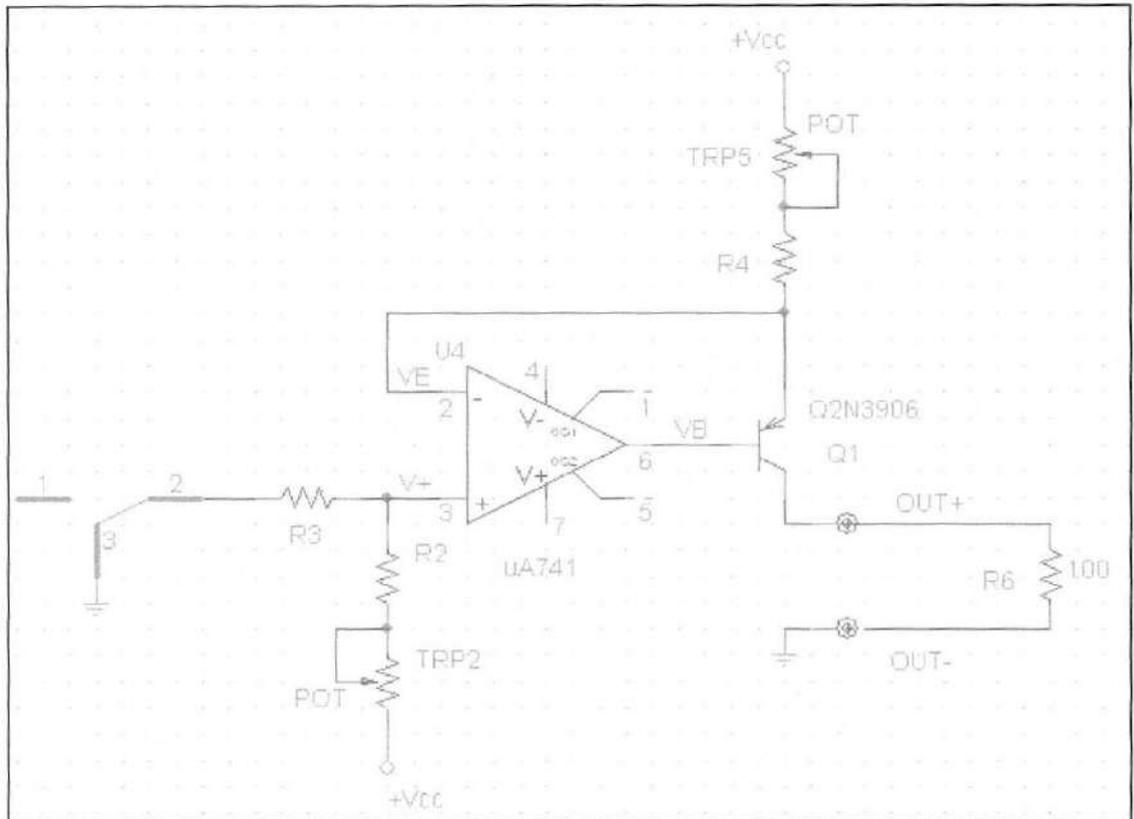


Figura 5.32.- Salida del circuito de la tarjeta transductora (jumper 2-3), que nos da una señal de 0 – 20mA.

1.- Ajuste  $I_{out} = 20 \text{ mA}$

Conectamos el jumper de la posición 3 a 2 y tenemos cero voltios en la entrada positiva del operacional U4 de la figura 5.32, correspondientes a 20mA en la salida para lo cual fijamos TRP5 y R4 aproximadamente en  $300\Omega$ . Luego

calibrando TRP2 ajustamos la salida en  $I_{out} = 20 \text{ mA}$ . En las borneras de salida de la tarjeta transductora conectamos una resistencia de  $100\Omega$ , como podemos ver en la figura 5.32, para cerrar el circuito y con un instrumento de medición en la salida controlar la calibración.

$$V_{+} = V_E = \frac{R_3}{R_3 + R_2 + TRP2} * V_{CC}$$

Ecuación 5.5.-

Asumiendo el operacional U4 en zona lineal  $V^{+} = V_E$  con TRP2 nos aseguramos que  $V_E > V_C$  del transistor Q1, según la ecuación 5.5, si  $V_{cc}$  es igual a 12V y  $R_2, R_3$  son iguales  $V_E$  será de hasta 6V con  $TRP2 = 0$ . Ajustando TRP2 obtendremos  $I_E \cong I_C = 20\text{mA}$  en el colector del transistor, con una entrada en el operacional U4 de cero Voltios, según la ecuación 5.6.

$$I_{out}(20\text{mA}) = \frac{V_{cc} - V_E}{R_4 + TRP5}$$

Ecuación 5.6.-

## 2.- Ajuste ISET (42° C), $I_{out} = 12 \text{ mA}$

Conectamos una resistencia externa R5 en las borneras de entrada de la tarjeta transductora según la Figura 5.33 para simular ISET del sensor en un rango entre 0-70°C conjuntamente con el ajuste de TRP1. La figura 5.34 muestra la función de transferencia de V02/V01 según la ecuación 5.7 donde observamos que V01 (señal en voltios correspondiente a ISET) se encuentra en un rango de 1,7-2,2 Voltios correspondiente a 0-70°C (ajuste TRP1). V02 es inversa y la pendiente de la curva es manejada por el parámetro TR1. El operacional U2 permite tener un rango de señal más amplio que V01 y por consiguiente de ISET debido a sus incrementos de 1  $\mu\text{A}/^\circ\text{K}$ . Con TR1 la curva de la Figura 5.34 se ajusta a la salida  $I_{out}$  correspondiente a la temperatura medida por el IC LM334Z.



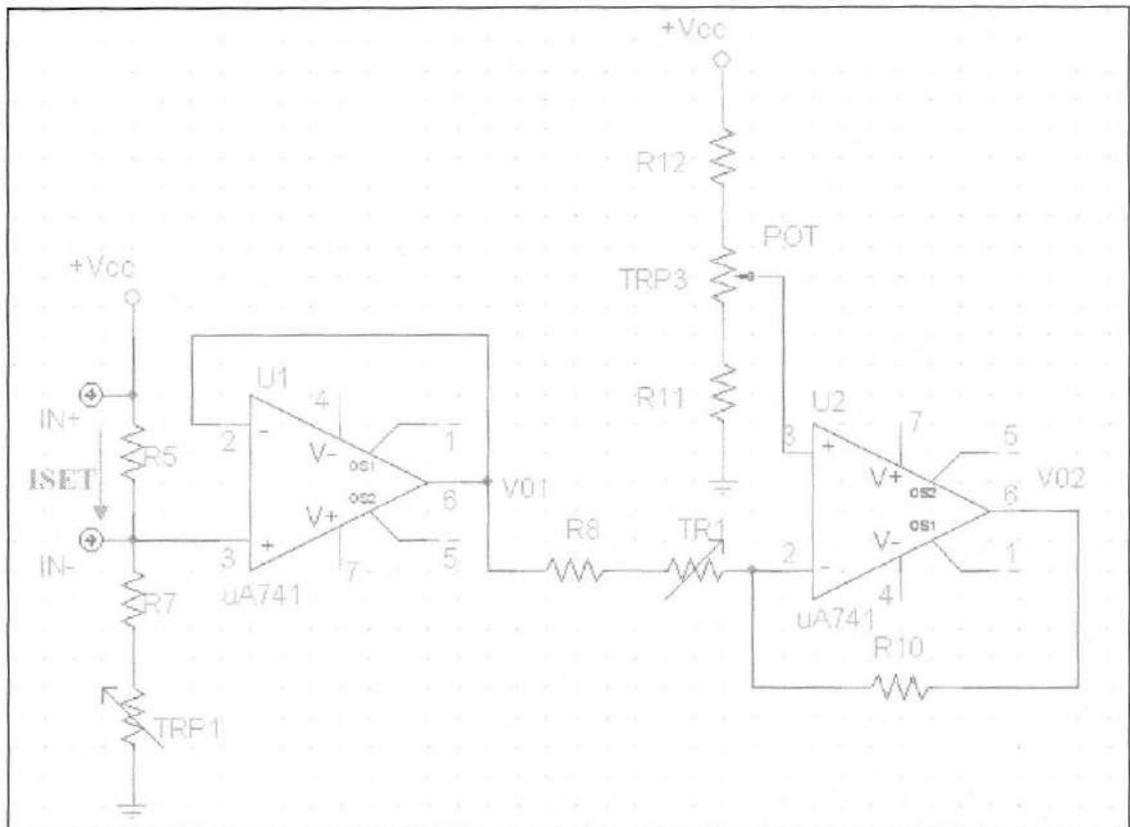


Figura 5.33.- Circuito de entrada de la tarjeta transductora, que nos da una señal de 0 – 20mA.

Simulamos ISET de 42°C (ecuación 5.4) con TRP1 hasta que en la salida de U1 (V01) exista una caída de voltaje de 2 voltios correspondiente a ISET (42°C). Luego con TRP3 ajustamos la salida del operacional U2 en  $V02 = 2V$  observamos en la Figura 5.34 que  $V01 = V02 = 2$  voltios es independiente del valor de TR1.

$$V_{O2} = \frac{R_{10}}{R_8 + TR1} \times (2 - V_{O1}) + 2$$

Ecuación 5.7.-

FUNCIÓN DE TRANSFERENCIA

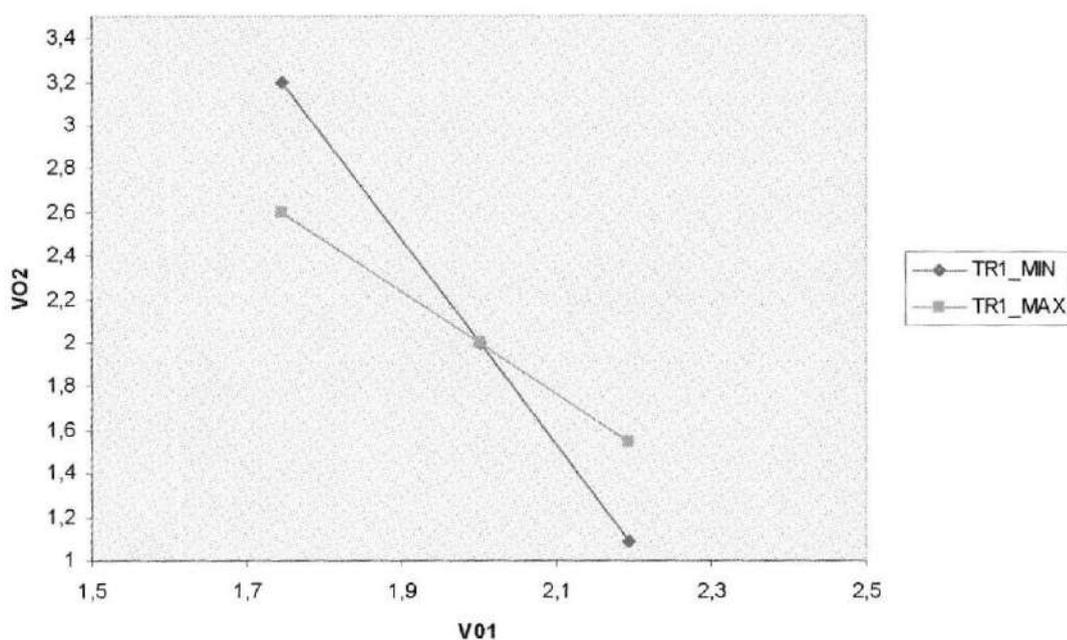


Figura 5.34.- Función de Transferencia entre V02 y V01, TR1 es el parámetro que ajusta la pendiente de la curva.

Luego de haber fijado 2 Voltios en la salida de U2 con TRP3 y V01 = 2V con TRP1, ponemos el jumper en la posición 1-2 como se ve en la figura 5.35. Luego con la tercera etapa (operacional U3) procedemos a ajustar TRP4 hasta

que la salida  $I_{out} = 12\text{mA}$  correspondiente a  $42^\circ\text{C}$ . La ecuación 5.8 muestra como se amplifica  $V_{02}$ .

$$V_{03} = \left( \frac{R9}{TRP4 + R8} + 1 \right) \times V_{02}$$

Ecuación 5.8.-

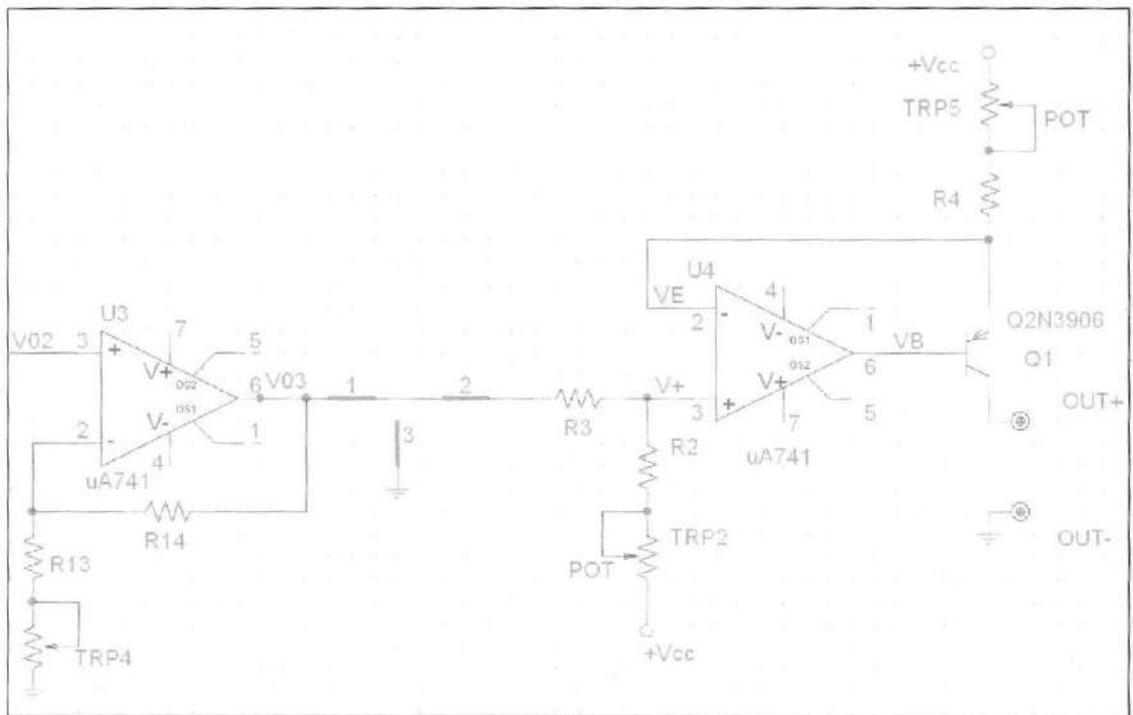


Figura 5.35.- Tercera etapa de amplificación de la tarjeta transductora 0-20mA.

3.- Ajuste ISET (0°C),  $I_{out} = 0\text{mA}$  y ISET (70°C),  $I_{out} = 20\text{mA}$

Una vez el jumper está en la posición 1-2, de la ecuación 5.5 obtenemos la ecuación 5.9 y con VE calculamos  $I_{out}$  (ecuación 5.6) la corriente de salida de la tarjeta.

$$V_{+} = V_{E} = \left( \frac{1}{R3 + R2 + TRP2} \right) \times (R3 * VCC + (R2 + TRP2) * V03)$$

Ecuación 5.9.-

El último paso está en ajustar el mínimo de  $V01 = 1,7\text{ V}$  (ISET 0°C) con  $I_{out} = 0\text{ mA}$  y el máximo de  $V01 = 2,2\text{ V}$  (ISET 70°C) con  $I_{out} = 20\text{mA}$  mediante el ajuste de TR1.

Una vez que se han hecho los ajustes en tarjeta transductora, retiramos la resistencia R5 y colocamos el sensor IC LM334Z en IN+ e IN- en la tarjeta.

Luego se procede en poner a punto la lectura del sensor por lo que se enumeran los siguientes pasos:

a) Medimos con un multímetro la señal de corriente (mA) del sensor IC LM334Z y calculamos la temperatura con la ecuación 5.4 y lo comparamos con la lectura de un termómetro de preferencia digital, en caso de no coincidir se hace el ajuste necesario con RSET (Figura 5.30).

b) Ajustamos TRP1 para que el rango de V01 (1.7-2.2V) corresponda al rango de 0 - 70°C de lectura del sensor IC LM334Z.

c) Calculamos  $I_{out}$  con la ecuación 5.10 con respecto a la señal V01 y medimos con un multímetro la corriente de salida. Comparamos los resultados y sino coinciden se hace el ajuste con TR1.

$$V01 = 22,5I_{out} + 1,74$$

Ecuación 5.10.-

d) Desconectamos la resistencia R6 de la tarjeta transdutora y su salida la conectamos a la tarjeta de adquisición de datos. Con la ecuación 5.11

calculamos el voltaje que ingresa al microcontrolador Pic de la tarjeta esclavo de acuerdo a la temperatura sino es así la calibramos con el trimmer ¿? (Anexo figura B-3) que se encuentra en la tarjeta de adquisición de datos (tarjeta esclavo) y la ecuación 5.12 (Ecuación 3.4 temperatura [°C]) es usada ya en el software para calcular la temperatura en °Kelvin y procedemos a comprobar lo que lee el módulo esclavo con lo que muestra el software sino coincide se ajusta el trimmer del circuito de la tarjeta esclavo.

$$V_{\text{temperatura}} = (5 / 20) I_{\text{out}}$$

Ecuación 5.11.-

$$T[{}^{\circ}K] = \left(\frac{70}{5}\right) V_{\text{temperatura}} + 273$$

Ecuación 5.12.-

### 5.3.2 MODULO DE CONVERSION ANALOGICO-DIGITAL.

El módulo de conversión Analógica-Digital (A/D), es un módulo propio del Pic. En este proyecto se usa un Firmware para controlarlo y ciertos periféricos externos para que funcione de una manera adecuada y a la aplicación del proyecto.

El módulo A/D es de 10 bits, con cinco canales analógicos de los cuales se usan cuatro:

- RA0, RA1, RA5 para las entradas de corriente, temperatura y voltaje.
- RA2 y RA3 como voltaje de referencia para la conversión.

Los periféricos que se usan para el módulo A/D, es un circuito que fija 5V en la entre RA3 y RA2 como se muestra en la figura 5.36, se trata de una resistencia y un diodo Zener en serie.



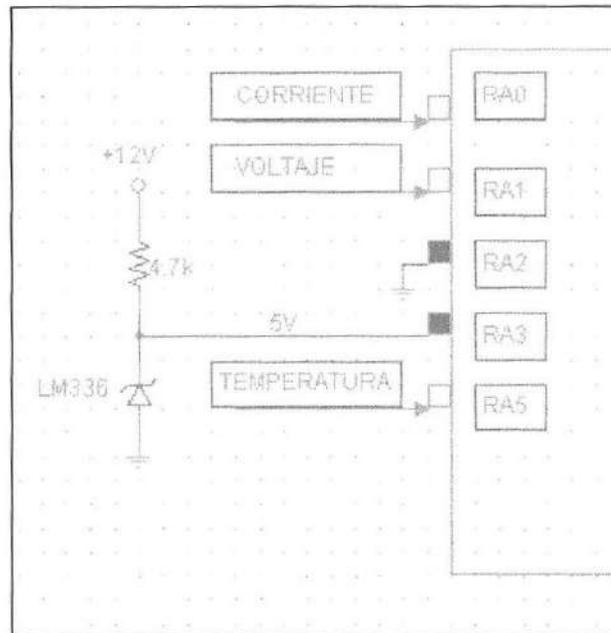


Figura 5.36.- Circuito que fija el voltaje de referencia en el Pic para la conversión.

En el Firmware del Pic se debe configurar el registro ADCON1 para tener como voltaje de referencia los pines RA3-RA2, y RA0, RA1, RA5 como entradas analógicas.

### *Operación*

Para la operación del A/D se realizó un módulo en el Firmware del Pic que controla dicho módulo. El Firmware configura los registros de control ADCON0 y

ADCON1. En el registro ADCON1 configura las entradas RA0, RA1, RA5 como analógicas y RA2, RA3 como entradas de referencia para la conversión, con los bits PCFG3-PCFG0. El bit ADFM se lo configura para justificar a la derecha el resultado de la conversión en los registros ADRESH: ADRESL ya que estos registros juntos guardan un número de 16 bits y el resultado de la conversión es de 10 bits. Mientras que el registro ADCON0 controla el funcionamiento del A/D.

Los bits ADCS1:ADCS0 establecen el tiempo de conversión por bit (TAD), en este caso establecido en  $F_{osc}/32$  como el Pic trabaja a una frecuencia de 20Mhz, TAD es de 1,6us y el tiempo de conversión de 10 bits total es de 11,5 TAD (18,4us) como se puede ver en la figura 5.37.

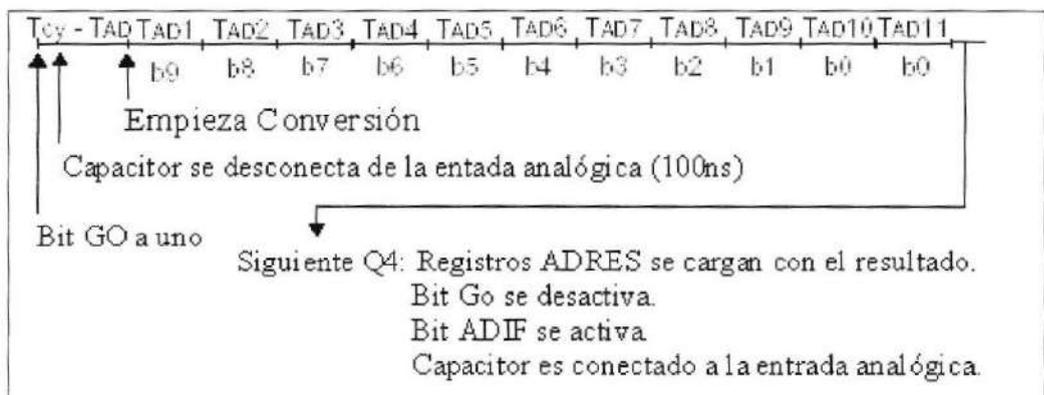


Figura 5.37.- Ciclos de conversión TAD.

Los bits CHS2:CHS0 seleccionan el canal analógico que va a ingresar al convertidor A/D, durante el Firmware se escoge el canal RA0, luego RA1 y RA5 en ese orden respectivamente. El bit GO inicia la conversión y ADON habilita el módulo A/D. El Diagrama de flujo de la figura 5.38, nos enseña la operación del módulo A/D. Cuando la conversión termina, GO se desactiva ( $GO = 0$ ) y el bit ADIF del registro PIR se activa.

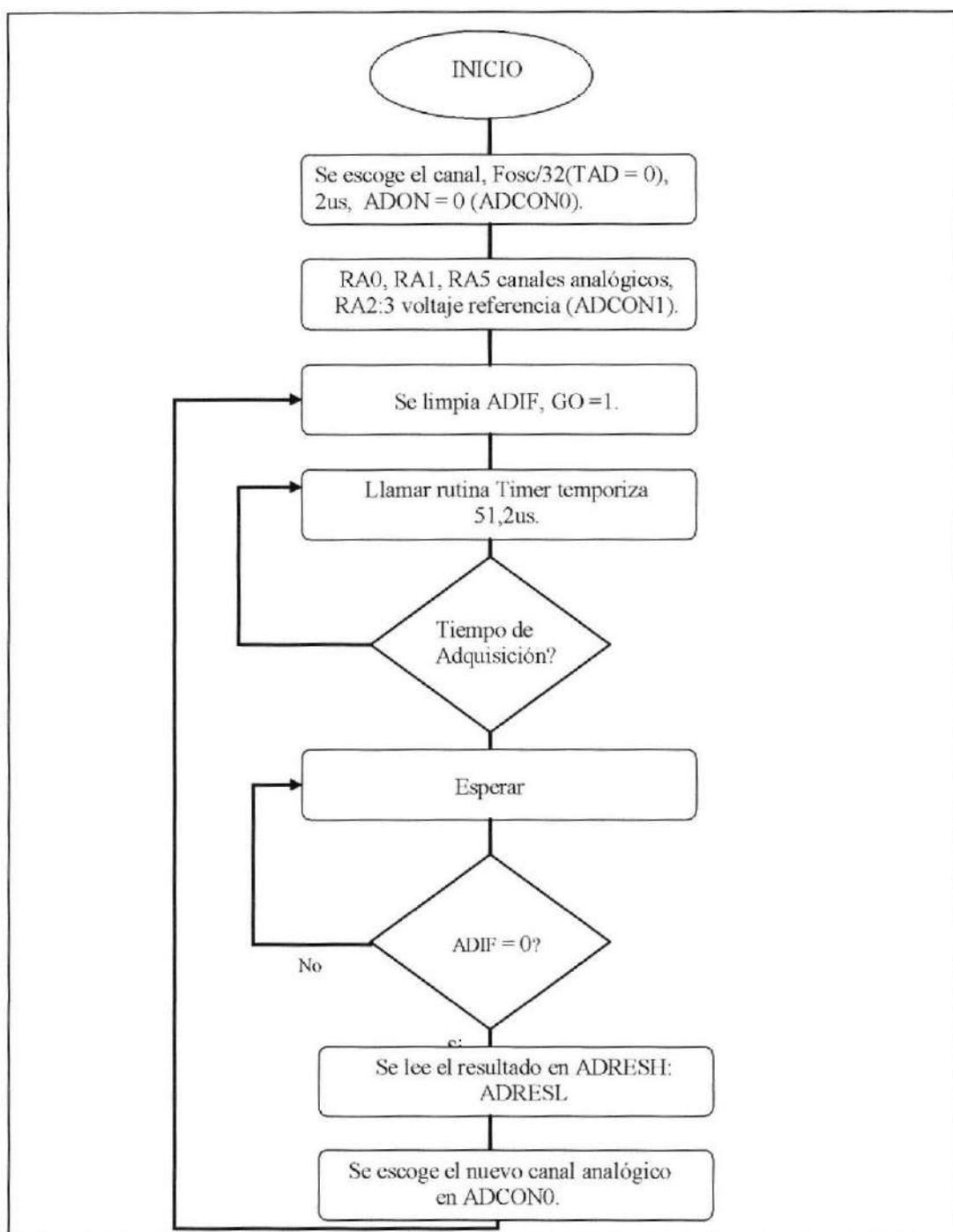


Figura 5.38. - Diagrama de flujos de la operación del módulo A/D.

### *Requerimientos*

Uno de los requerimientos es el tiempo de adquisición del canal analógico, el cual es el tiempo que el módulo A/D conecta el capacitor "sample and hold" al voltaje analógico externo, la deducción de este tiempo se encuentra en el capítulo 4. Si este tiempo no se cumple el resultado de la conversión no será el correcto. Otro tiempo requerido es el tiempo de conversión de un nivel analógico a un valor digital de 10 bits. El tiempo de conversión por bit (TAD) para un Pic a 20MHZ es de 1.6us, por lo tanto como el tiempo de conversión es 11,5 TAD, entonces este tiempo será de 18,4us.

La suma de estos dos tiempos es el tiempo de muestreo que hay que respetar cuando se pasa de una conversión a otra. El tiempo de adquisición usado en el proyecto es de 51,2us, tiempo más que suficiente para la adquisición de los canales analógicos y para el muestreo de las señales que se explica en las secciones siguientes.

Además la impedancia de entrada a los canales es máximo de  $10\text{ k}\Omega$ , si este valor es menor el tiempo de adquisición podrá ser menor.

Si el voltaje en la entrada analógica excede el valor de  $V_{ss}$  o  $V_{dd}$ , por encima de  $0,3V$ , entonces la exactitud de la conversión estará fuera de las especificaciones.

En el circuito del tratamiento de la corriente de la figura 5.27 se puso una configuración RC para mejorar la estabilización de la entrada para que no introduzca un error en el A/D debido a las características de la corriente vistas en la sección anterior. La componente de la resistencia debería ser seleccionada tal que la impedancia de entrada se mantenga por debajo de  $10\text{ k}\Omega$  recomendado por las especificaciones. Además todo componente conectado a una entrada analógica (capacitor, zener, etc) debería tener poca corriente de fuga en dicha pin de entrada.

### Resolución, Exactitud y Error

El Pic usado para el sistema de monitoreo da un resultado digital de 10 bits, según la ecuación 5.13 y con  $V_{ref} = 5V$ , equivale a que cada 4mV analógicos corresponde a un bit (4mV/bit).

$$Resolución = \frac{(V_{ref} - V_{ref-})}{1024} = \frac{V_{ref}}{1024}$$

Ecuación 5.13

La resolución y exactitud de los datos que se monitorean en el sistema son de mucha importancia, dependiendo de ellos se manejan muchos criterios en el software de monitoreo. Datos incorrectos pueden asumir criterios incorrectos, por lo tanto conclusiones falsas con respecto al ciclo de carga de las líneas de baterías.

La rapidez de la conversión depende de TAD (tiempo de conversión por bit), y este a la vez depende del oscilador del Pic, en nuestro caso nuestro oscilador es de 20Mhz, por lo cual consideramos nuestra conversión rápida, además el tiempo de conversión es un parámetro de relativa importancia ya que si bien se deben tomar los datos de algunos cargadores de una sola vez también los procesos de carga son mayores a 36 horas en unos casos.

La exactitud del módulo A/D incluye la suma de todas las contribuciones del error de cuantización, error integral, error diferencial, error de escala completa, error offset. Esto es definido como la máxima desviación de una transición actual versus una transición ideal para cualquier código digital. El error absoluto del A/D es de  $< \pm 1LSb$  de  $V_{dd} = V_{ref}$ . Sin embargo, la exactitud del A/D puede degradarse así como  $V_{dd}$  diverge de  $V_{ref}$ .

Para un rango de voltajes analógicos en la entrada, y debido a la resolución de 4mV/bit del A/D, cambios en la entrada menores de 4mV tendrá como resultado la misma salida digital, a esto se le llama cuantización de una entrada digital a un código digital. El error de cuantización por lo general es de  $\pm 1/2LSb$

y es inherente a un proceso de conversión analógico a digital. La única manera de reducir el error de cuantización es incrementar la resolución del convertidor A/D. El error Offset mide la primera transición actual de un código versus la primera transición ideal de un código. El error Offset cambia la función de transferencia completa de la figura 5.39, puede ser calibrado fuera de un sistema (en el mismo Firmware del Pic) o introducido dentro de un sistema a través de la iteración de la corriente de fuga total y la fuente de impedancia en la entrada analógica. El error de ganancia mide la máxima desviación de la última transición actual y la última transición ideal ajustada por el error Offset. Este error aparece como un cambio en la pendiente de la función de transferencia. La diferencia entre el error de la ganancia y el error de escala completa es que el de escala completa no lleva el error Offset en consideración. El error de ganancia puede ser calibrado en software. El error Offset y de ganancia pueden ser corregidos vía software en el Firmware del Pic.

El error de linealidad se refiere a la uniformidad de los cambios en el código. El error de linealidad no puede ser calibrada fuera del sistema. El error de no linealidad integral mide que tan cerca está la actual transición de un código de la

transición ideal de un código ajustado por el error de ganancia para cada código.

El error de no linealidad diferencial mide el actual ancho de código versus el ideal ancho de código. Estos dos errores son difíciles de corregir en el software, ya que para cada código debería ser evaluado para una calibración adecuada. En pocas palabras se manifiesta como ruido, una manera para eliminarla es usar un filtro antes de la entrada, esto disminuiría estos ruidos.

Para este proyecto, se usan Operacionales que ayudan a la ganancia de amplificación de la señal y aseguran una impedancia de salida baja conectada a la entrada del convertidor, asegurando un tiempo de conversión rápido. El operacional contribuye con cierto ruido que es tratado en el Firmware del Pic, se realiza un muestreo de la señal, que es tratada como si fuera una señal AC, la frecuencia de muestreo es aproximadamente de 20Khz, y luego se obtiene un promedio, esto se explica en la siguiente sección.



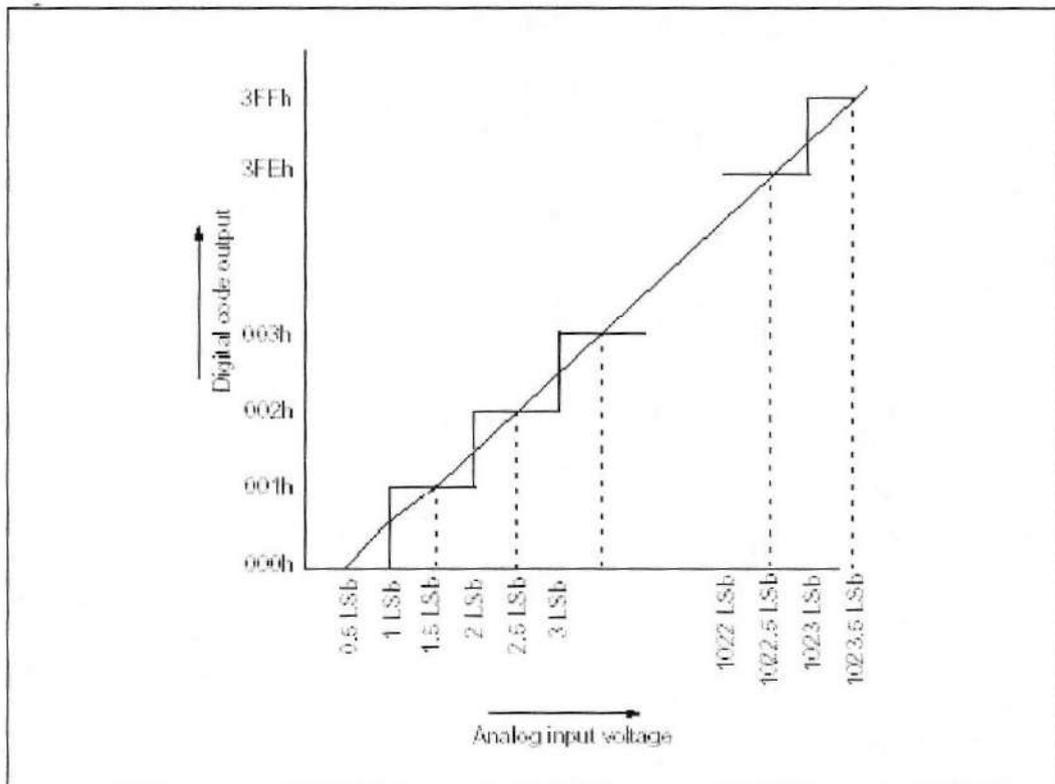


Figura 5.39.- Función de transferencia.

### 5.3.3 MODULO DE MUESTREO Y PROMEDIO DE SEÑALES ANALOGICAS.

#### *Muestreo De Señales*

El módulo Esclavo realiza un muestreo de las señales que se están monitoreando, ya que por diversos factores las lecturas de los datos variarían,

como lo son el error que aporta el convertidor A/D y el tipo de señal que se monitorea. Como ya vimos en las secciones anteriores el tipo de señal de la corriente (Figura 5.24) no es completamente DC pura y aproximadamente con una frecuencia de 120Hz, mientras que la frecuencia de conversión del modulo A/D es de 19,5Khz por lo que aproximadamente en un período se podrían tomar 163 datos aproximadamente. El módulo esclavo toma 255 datos en un período y medio de onda aproximadamente. Este muestreo se realiza para la señal de corriente, voltaje y temperatura pero es más crítico para la señal de corriente ya que viene en milivoltios, y viene con un error del Opamp y del módulo conversor A/D, en las otras señales no es tan crítico ya que la señal que entra a la tarjeta es en el orden de los voltios, pero también se hace un muestreo para tener una mayor fiabilidad de los datos luego del proceso de conversión.

La señal de corriente monitoreada por el módulo esclavo se ve en la figura 5.40.

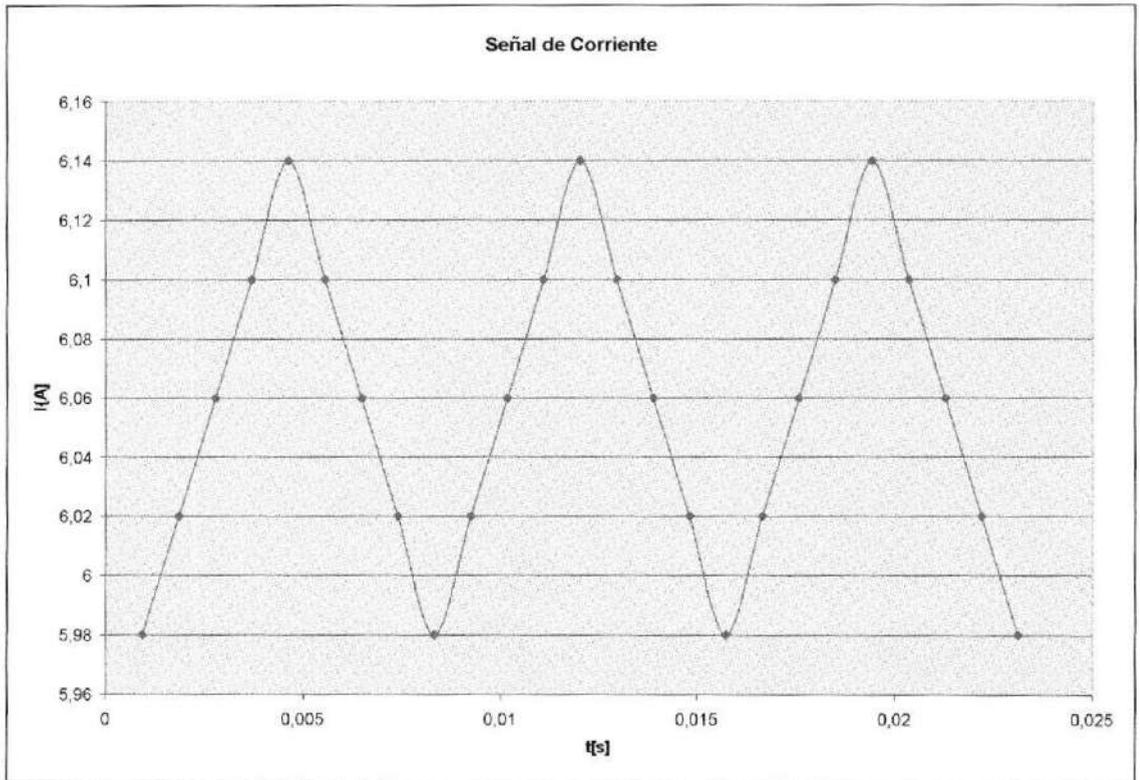
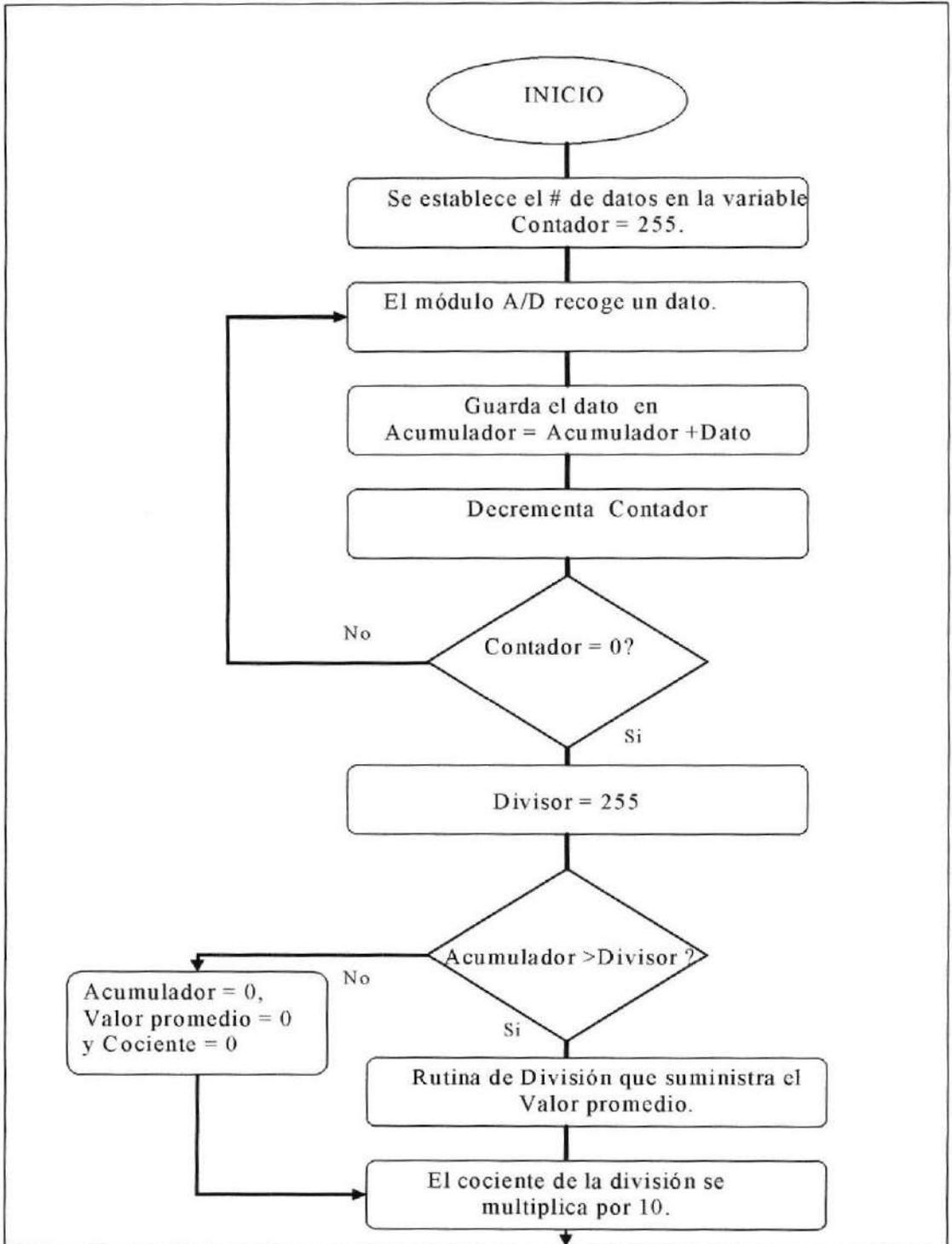


Figura 5.40.- Señal de corriente que muestrea el módulo esclavo a una frecuencia de 19,5 KHz.

### *Promedio de señales*

Una vez las señales son monitoreadas pasamos por el módulo de promedio de señales analógicas donde obtenemos el valor promedio de la señal, que es enviada a la PC.

El promedio es de 255 datos y el módulo tiene un Firmware que lo controla. El diagrama de flujos de los módulos de muestreo y del promedio se encuentra en la figura 5.41.



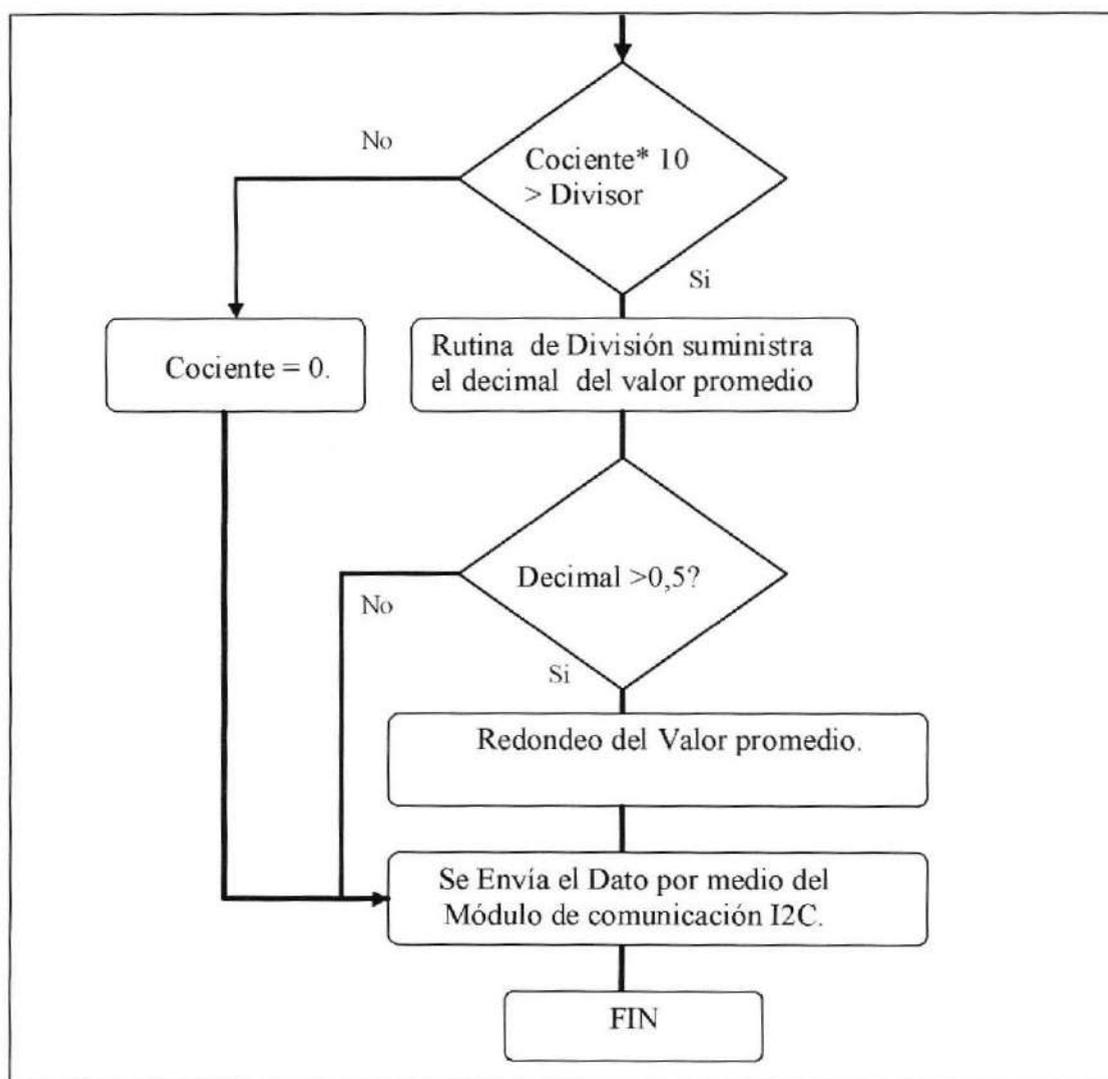


Figura 5.41.- Diagrama de flujos del módulo de muestreo y promedio de señales analógicas.

### 5.3.4 MODULOS ESPECIALES

#### *1.- Modulo de conexión al sistema*

El sistema de monitoreo de baterías implementado consta de dos tarjetas esclavos y una tarjeta maestro. Las tarjetas esclavo se encuentra en cada salida o línea de un cargador. De las salidas (líneas) de los cargadores se obtienen los datos de voltaje y corriente. La línea positiva del cargador es la referencia analógica del módulo de amplificación y acondicionamiento de señales analógicas, debido a la posición del shunt (Figura 5.28) y la conexión de la línea baterías. Por otra parte existe una referencia digital que corresponde al microcontrolador de la tarjeta esclavo. Todas las referencias digitales de todos los esclavos están unidas entre sí y con la referencia digital del maestro y este último con la de la PC. Mientras que las referencias analógicas se encuentran aisladas entre sí y con la PC. Al momento en que la PC realiza la petición de datos, el maestro direcciona módulos esclavos, cuando éstos responden, el módulo de conexión al sistema del esclavo une la referencia analógica con la referencia digital del módulo y con la referencia digital de todo el sistema (Figura 5.42). De esta forma el maestro recoge datos de los esclavos. Cada esclavo

tiene un módulo de conexión al sistema que conecta la parte analógica del esclavo con el sistema cada vez que es direccionado. En la figura 5.42 se puede observar el esquema de conexiones entre las tarjetas esclavos y la tarjeta maestro. En el Pic16F877A se encuentra el firmware del módulo de conexión del sistema que activa el relé al ser direccionado el esclavo, conectando la parte analógica del Esclavo al sistema.

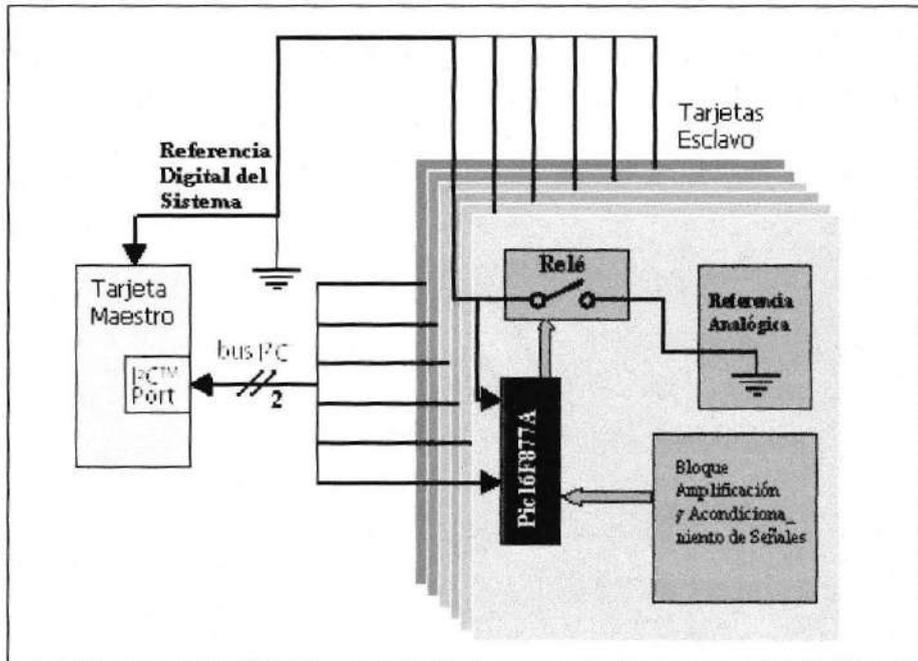


Figura 5.42.- Esquema de conexiones del Sistema de Monitoreo, tarjeta Maestro con tarjetas Esclavos.

## 2.- Módulo Perro Guardián (WDT)

El Bloque del Perro guardián, es un bloque especial que el firmware usa en lugares específicos de la ejecución del programa interno del microcontrolador para que ninguna parte se quede colgado el firmware. Se encuentra tanto en el firmware Maestro y Esclavo, pero su aplicación es más vital en la firmware esclavo por el procesamiento de los datos en la conversión, suma, promedio y comunicación de datos. Se configura el bloque WDT asignando el bit PSA (OPTION\_REG<3>) a uno, predivisor asignado al WDT, y los bits PS2:PS0 (OPTION\_REG<2:0>) en 111 asignando un predivisor de 128, con lo que el WDT no es nada más que un contador que al desbordarse provoca un Reset, el tiempo en que se desborda con el predivisor asignado es de 2,3 segundos aproximadamente y provoca que el microcontrolador despierte del modo de bajo consumo o SLEEP y continúe con la ejecución del programa. Pero el firmware detecta si se ha despertado del SLEEP monitoreando los bits TO (STATUS<4>) y PD (STATUS<3>). Si los bits PD y TD están en bajo, significa que el microcontrolador ha despertado del SLEEP por desbordamiento del WDT y el firmware reinicia el programa, pero sino continúa con la ejecución del mismo. Pero puede existir un desbordamiento del WDT cuando no es requerido

en ciertas partes del firmware como por ejemplo a la espera de alguna entrada lógica por uno de las puertas de Entrada/Salida, por lo tanto en estas partes del firmware se refresca (Limpia) el WDT con la instrucción "CLRWDT".

## CONCLUSIONES Y RECOMENDACIONES.

La razón de un sistema de monitoreo no es más que la de llevar un control de un proceso, en la fabricación de baterías. El sistema es muy útil sobretodo en una de las últimas partes del proceso y una de las más importantes, la formación eléctrica de baterías, ya que el control en estos procesos de carga son a menudo muy largos, aproximadamente 36 horas (Según el fabricante) y se necesita hacer uso de personal de supervisión de no menos de tres personas en varios turnos para llevar tal control. Mientras que el Sistema de Monitoreo podría hacer ese trabajo únicamente con revisiones periódicas de cada 3 o 4 horas al sistema y sus archivos del proceso de carga. Sobretodo en casos en que una batería regrese por encontrarse defectuosa, los archivos del sistema permitirían revisar el historial de carga de esa batería y así poder determinar la causa del problema, además permitiendo las rectificaciones del caso en futuros procesos de carga.



CIB-ESPOL

Combinando la electrónica para obtener las señales, los microcontroladores para el procesamiento y comunicación con una PC y las herramientas de la

programación en Visual Basic para administrar la información e interacción con el usuario final, el Sistema de monitoreo se convierte en una verdadera bitácora electrónica del proceso de formación eléctrica de baterías.

Las recomendaciones del caso es que el Sistema podría ser más completo y eficiente si podría tomar decisiones y acciones sobre el cargador que forma la batería, convirtiéndolo en un sistema de Monitoreo Inteligente realimentado por nuevos programas de carga

# ANEXOS

## A. ESPECIFICACIONES ELECTRICAS DE LOS ELEMENTOS DEL SISTEMA.

TABLA A-1

CARACTERISTICAS DC DEL PIC16F877A

Pic 16F877A						
Símbolo	Características	Min.	Tip.	Max.	Unid.	Condición
VDD	Fuente de Voltaje	4.5	—	4.5	V	Configuración HS osc.
VDR	Voltaje de detención de datos RAM.	—	1.5	—	V	
VPOR	Voltaje de Reset por conexión a la alimentación.	—	V <sub>SS</sub>	—	V	
SVDD	Tasa de elevación del Voltaje.	0.05	—	—	V/ms	
VBOR	Voltaje de caída de Tensión.	3.7	4.0	4.5	V	Bit BODEN habilitado en la palabra de configuración.
IDD	Fuente de Corriente	—	7	15	mA	Configuración HS osc. Fosc = 20 MHz, VDD = 5.5V
ΔIBOR	Corriente de caída de Tensión BOR.	—	85	200	μA	Habilitado BOR, VDD = 5.0V

TABLA A-2.

CARACTERISTICAS DC DEL PIC16F877A.

Pic 16F877A						
Símbolo	Características	Min.	Tip.	Max.	Unid	Condición
VIL	<b>Nivel bajo de Voltaje de entrada.</b>					
	Puertos I/O con buffer TTL	Vss	—	.15Vdd	V	Para rangos entre 4.5V = VDD = 5.5V
		Vss	—	0.8V	V	
	con Schmitt Trigger	Vss	—	0.2Vdd	V	
	MCLR, OSC1 (en modo RC)	Vss	—	0.2Vdd	V	
	OSC1 (en XT, HS y LP)	Vss	—	0.3Vdd	V	
Puerto RC3 y RC4 Con Schmitt Trigger Con SMBus	Vss -0.5	— —	0.3Vdd 0.6	V V	Para rangos de VDD = 4.5 a 5.5V.	
VIH	<b>Nivel alto de Voltaje de entrada.</b>					
	Puertos I/O con TTL buffer	2.0	—	Vdd	V	Para rangos entre 4.5V = VDD = 5.5V
		.25Vdd	—	Vdd	V	
	con Schmitt Trigger	+ 0.8V	—	Vdd	V	
	MCLR	0.8Vdd	—	Vdd	V	
	OSC1 (XT, HS y LP)	0.7Vdd	—	Vdd	V	
Puertos RC3 y RC4 Con Schmitt Trigger Con SMBus	0.7Vdd 1.4	— —	Vdd 5.5	V V	Para rangos entre VDD = 4.5 a 5.5V	

IPURB	<b>Corriente en Puerto B por resistencias Pull-up.</b>	50	250	400	$\mu\text{A}$	Vdd = 5V, VPIN = VSS, 40°C to +85°C
III	<b>Corriente de Fuga En la entrada.</b>					
	Puertos I/O	—	—	$\pm 1$	$\mu\text{A}$	Vss $\leq$ VPIN $\leq$ Vdd, Pin en alta impedancia.
	MCLR, A4/T0CKI OSC1	—	—	$\pm 5$ $\pm 5$	$\mu\text{A}$ $\mu\text{A}$	Vss $\leq$ VPIN $\leq$ VDD Vss $\leq$ VPIN $\leq$ VDD, Configuración XT, HS y LP Osc.

TABLA A-3

CARACTERISTICAS DC DEL PIC16F877A.

Pic 16F877A						
Símbolo	Características	Min.	Tip.	Max.	Unid.	Condición
VOL	<b>Nivel bajo de Voltaje de entrada.</b>					
	Puertos I/O con buffer TTL	Vss	—	0.15VDD	V	Para rangos entre 4.5V = VDD = 5.5V
		Vss	—	0.8V	V	
	con Schmitt Trigger MCLR, OSC1(en modo RC)	Vss	—	0.2VDD	V	
		Vss	—	0.2VDD	V	
	OSC1 (XT, HS, LP) Puerto RC3 y RC4	Vss	—	0.3VDD	V	
Con Schmitt Trigger Con SMBus	Vss -0.5	—	0.3VDD 0.6	V V	Para rangos de VDD = 4.5 a 5.5V.	
VOH	<b>Nivel alto de Voltaje de entrada.</b>					

	Puertos I/O con TTL buffer	2.0 .25VDD + 0.8V	— — —	VDD VDD	V V	Para rangos entre 4.5V = VDD = 5.5V
	con Schmitt Trigger MCLR OSC1 (XT,HS,LP) Puertos RC3 y RC4 Con Schmitt Trigger Con SMBus	0.8VDD 0.8VDD 0.7VDD 0.7VDD 1.4	— — — — —	VDD VDD VDD VDD 5.5	V V V V V	Para rangos entre VDD = 4.5 a 5.5V
VOD	<b>Corriente Puerto B por resistencias Pull-up.</b>	50	250	400	µA	VDD = 5V, VPIN = VSS, 40°C to +85°C
<b>Especificaciones de Capacitancia en los pines de salida.</b>						
COSC2	OSC2 pin	—	—	15	pF	En modos XT,HS y LP Cuando un reloj externo es usado para manejar OSC1.
CIO	All I/O pins and OSC2 (RC mode)	—	—	50	pF	
CIB	SCL, SDA (I2C)	—	—	400	pF	
<b>Memoria de datos EEPROM</b>						
ED	Durabilidad	100K	—	—	B/E	25°C a 5V
VDRW	VDD para Lectura/Escritura	VMIN	—	5.5	V	Usando EECON para lectura/escritura.
TDEW	Ciclo Borrado/Escritura	—	4	8	ms	VMIN = min. Voltaje de operación.
<b>Memoria de Programa FLASH.</b>						
EP	Durabilidad	1000	—	—	E/W	25°C a 5V
VPR	VDD para lectura	VMIN	—	5.5	V	VMIN = min. voltaje de operación.
TPEW	VDD para B/E	VMIN	—	5.5	V	Usando EECON para lectura/escritura.
	Ciclo de Borrado/Escritura	—	4	8	ms	VMIN = min. voltaje de operación

FIGURA A-1

TIEMPOS DE RESET, PERRO GUARDIAN (WATCHDOG TIMER), OST (OSCILLATOR START-UP -TIMER) Y PWRT (POWER-UP-TIMER).

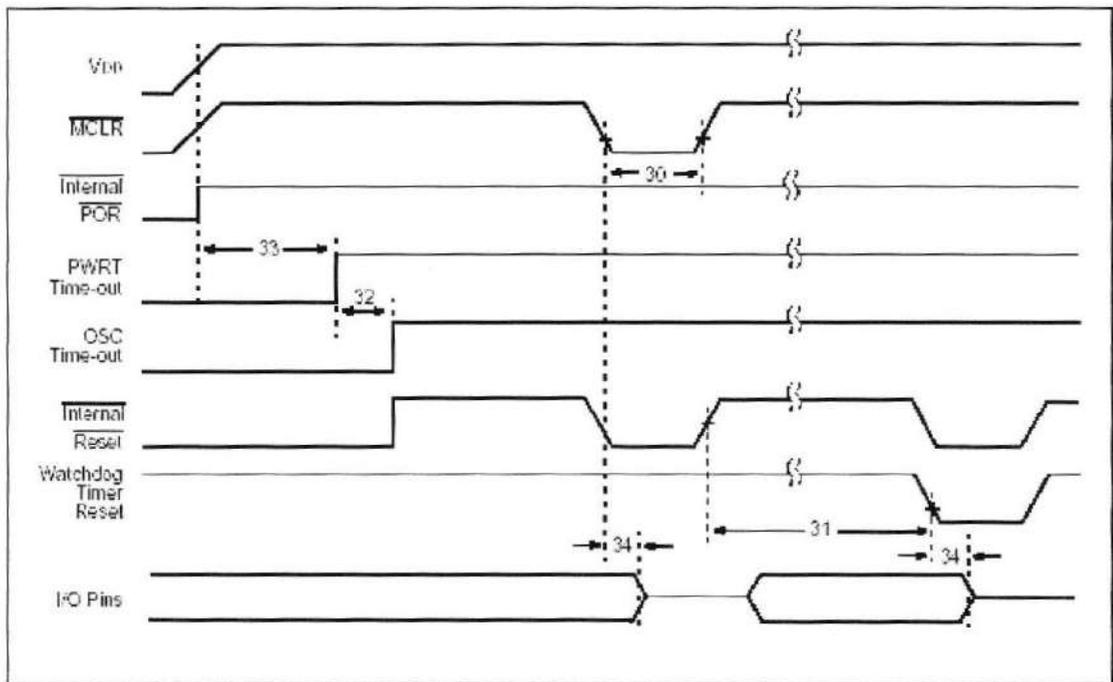


FIGURA A-2

TIEMPOS DE RESET POR FALLO EN LA ALIMENTACION BOR (BROWN-OUT).

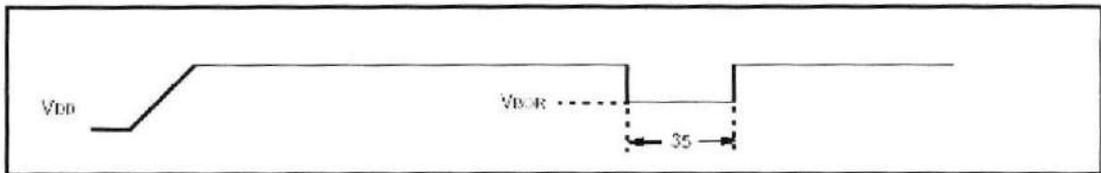


TABLA A-4

REQUERIMIENTOS DE RESET, PERRO GUARDIAN (WATCHDOG TIMER), OST (OSCILLATOR START-UP -TIMER) Y PWRT (POWER-UP-TIMER). DEL PIC16F877A.

Pic 16F877A						
Símbolo	Características	Min.	Tip.	Max.	Unid.	Condición
Tmcl (30)	Ancho de Pulso en bajo MCLR.	2	—	—	μs	VDD = 5V, -40°C a +85°C
Twdt (31)	Watchdog Timer Período de WDT (No Predivisor)	7	18	33	ms	VDD = 5V, -40°C a +85°C
Tost	Período de OST	—	1024 TOSC	—	—	TOSC = OSC1 período
Tpwrt (32)	Power-up Timer Period de PWRT	28	72	132	ms	VDD = 5V, -40°C a +85°C
TIOZ (33)	E/S alta impedancia por MCLR en bajo o reset del WDT.	—	—	2.1	μs	VDD. VBOR (D005)
TBOR (34)	Ancho de pulso del BOR (Brown Out).	100	—	—	μs	VDD = 5V, -40°C a +85°C

TABLA A-5

CARACTERISTICAS ELECTRICAS DEL MAXIM 232-A.

MAXIM 232-A					
Parámetros	Min.	Tip.	Max.	Unid	Condición
Fuente de Voltaje (VCC)	0.3V		+6V	V	
Voltajes de Entrada	0.3V		VCC - 0.3V	V	
TIN	-30		+30V	V	
RIN	-15		+15V	V	
TOUT	-15		+15V	V	
Voltajes de Salida	0.3V		VCC + 0.3V	V	
TOUT	0.3V		+6V	V	
ROUT	-0.3V		VCC - 0.3V	V	
Disipación de Potencia continua (TA = +70°C)			842	mW	
<b>Transmisor RS-232</b>					
Umbral Entrada lógica en Bajo.	±5	1.4	0.8	V	Todas las salidas transmisoras con una carga 3k½ a GND
Variación del Voltaje de Salida	2	±8	0.4	V	
Umbral entrada lógica en Alto.		1.4		V	
Entrada lógica Corriente		5	40	µA	Normal operation
Fuga de Corriente Salida		±0.01	±10	µA	Vcc = 5.5V, SHDN = 0V, Vout = ±15V
Tasa Transmisión datos		200	116	Kbits/sec	Normal Operación
Resistencia de la salida transmisor	300	10M		Ω	Vcc = V+ = V- = 0V, Vout = ±2V.
Salida Corriente corto circuito.	±7	±22		mA	Vout = 0V
<b>Receptor RS-232</b>					
Rango Operación entrada Voltaje RS-232.			±30	V	
Umbral de entrada RS-232 en bajo.	0.8	1.3		V	Vcc = 5V

Umbral de entrada RS-232 en alto.		1.8	2.4	V	$V_{cc} = 5V$
Histéresis de entrada RS-232	0.2	0.5	1	V	$V_{cc} = 5V$
Resistencia de entrada RS-232	3	5	7	$K\Omega$	
Salida de Voltaje TTL/CMOS en bajo.		0.2	0.4	V	$I_{out} = 3.2mA$
Salida de Voltaje TTL/CMOS en alto.	3.5	$V_{cc} - 0.2$		V	$I_{out} = -1.0mA$
Salida de corriente TTL/CMOS en corto circuito.	-2	-10		mA	$V_{out} = GND$
	10	30			$V_{out} = V_{cc}$
Salida de Corriente TTL/CMOS en Fuga.		$\pm 0.05$	$\pm 10$	$\mu A$	$Shdn = V_{cc}, EN = V_{cc}, 0V \leq V_{out} \leq V_{cc}$

FIGURA A-3

TIEMPOS DE LA TRASMISION DEL RS-232A.

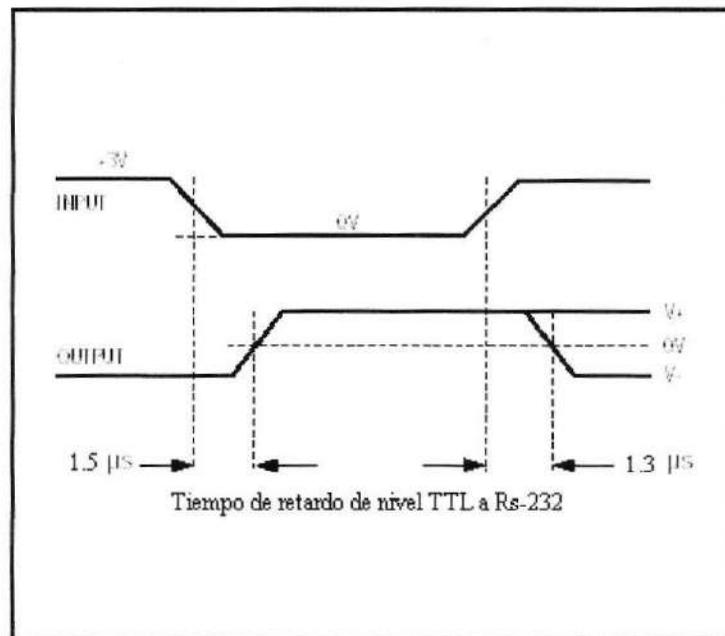
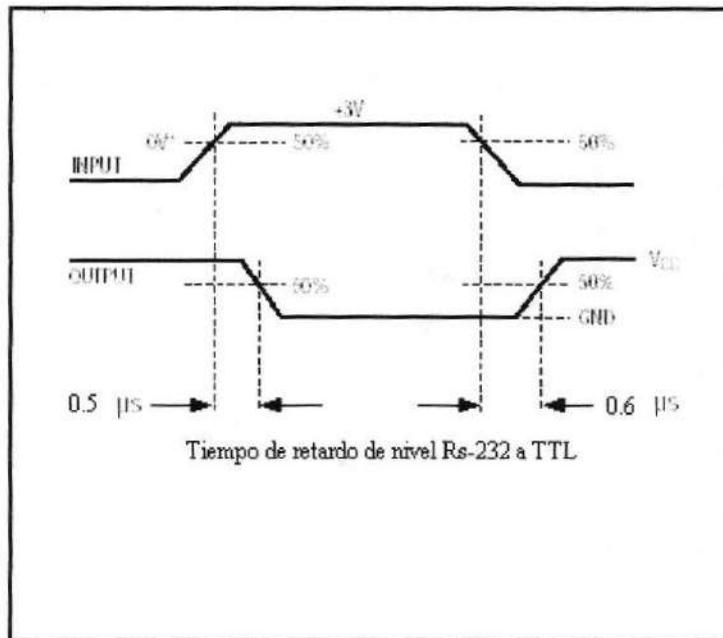


FIGURA A-4  
 TIEMPOS DE LA RECEPCION DEL RS-232A.



*RANGOS MAXIMOS DE OPERACION DEL LM334Z*

Voltaje V+ a V-	40V
Voltaje inverso V+ a V-	20V
Voltaje de Pin R a V-	5V
Corriente Iset	10 mA
Disipación de Corriente	400 mW
Rango Temperatura de Operación	0°C a +70°C

TABLA A-6

CARACTERISTICAS ELECTRICAS DEL LM334Z.

LM334Z					
Características	Min.	Tip.	Max.	Unid.	Condición
Error Corriente Iset V+ = 2.5V.	—	—	6	%	$10\mu\text{A} \leq \text{ISET} \leq 1\text{mA}$
	—	—	8	%	$1\text{mA} < \text{ISET} \leq 5\text{mA}$
	—	—	12	%	$2\mu\text{A} \leq \text{ISET} < 10\mu\text{A}$
Radio Iset a Corriente BIAS.	14	18	26		$100\mu\text{A} \leq \text{ISET} \leq 1\text{mA}$
	—	14	—		$1\text{mA} \leq \text{ISET} \leq 5\text{mA}$
	—	18	26		$2\mu\text{A} \leq \text{ISET} \leq 100\mu\text{A}$
Voltaje mínimo de operación.	—	0.8	—	V	$2\mu\text{A} \leq \text{ISET} \leq 100\mu\text{A}$
	—	0.9	—	V	$100\mu\text{A} < \text{ISET} \leq 1\text{mA}$
	—	1.0	—	V	$1\text{mA} < \text{ISET} \leq 5\text{mA}$
Cambio Promedio en Corriente Iset.	—	0.02	0.1	%/V	$2\mu\text{A} \leq \text{ISET} \leq 1\text{mA}$
	—	0.01	0.05	%/V	$1.5 \leq V+ \leq 5\text{V}$
	—	0.03	—	%/V	$5\text{V} \leq V+ \leq 40\text{V}$
	—	0.02	—	%/V	$1\text{mA} < \text{ISET} \leq 5\text{mA}$
Dependencia de Temperatura de la Corriente Iset.	0.96T	T	1.04T		$1.5\text{V} \leq V \leq 5\text{V}$
					$5\text{V} \leq V \leq 40\text{V}$
Capacitancia Efectiva Shunt.	—	15	—	pf	

RANGOS MAXIMOS DE OPERACION DEL  $\mu$ A7805 Y  $\mu$ A7812.

Entrada de Voltaje	35 V
Impedancia Térmica Juntura-Ambiente $\theta_{JA}$	25°C/W
Operación en Temperatura de Juntura $T_J$	150°C

TABLA A-7

CARACTERISTICAS ELECTRICAS DEL  $\mu$ A7805 PARA  $V_I = 10V$ ,  $I_O = 500mA$ .

$\mu$ A7805						
Características	$T_J$	Min.	Tip.	Max.	Unid	Condición
Salida de Voltaje	25°C	4.8	5	5.2	V	IO = 5 mA a 1 A, VI = 7 a 20V, PD $\leq$ 15 W
	0 a125°C	4.75	—	5.25		
Regulación de entrada de Voltaje	25°C	—	3	100	mV	VI = 7 V a 25 V
		—	1	50		VI = 8 V a 12 V
Rizado Rechazado	0 a125°C	62	78	—	dB	VI = 8 V a 18 V, f = 120 Hz
Regulación de salida de Voltaje	25°C	—	15	100	mV	IO = 5 mA a 1.5 A
		—	5	50		IO = 250 mA a 750 mA
Resistencia Salida	0 a125°C	—	0.017	—	$\Omega$	f = 1 kHz
Coefficiente de Temperatura de la salida de Voltaje	0 a125°C	—	-1.1	—	mV/ °C	IO = 5 mA
Ruido de Salida	25°C	—	40	—	$\mu$ V	f = 10 Hz a 100 kHz
Voltaje Dropout	25°C	—	2	—	V	IO = 1 A
Corriente Bias	25°C	—	4.2	8	mA	
Cambio de corriente Bias	0 a125°C	—	—	1.3	mA	VI = 7 V a 25 V
		—	—	0.5		IO = 5 mA a 1 A
Corriente Corto circuito de salida	25°C	—	750	—	mA	
Pico de Corriente de salida	25°C		2.2	—	A	

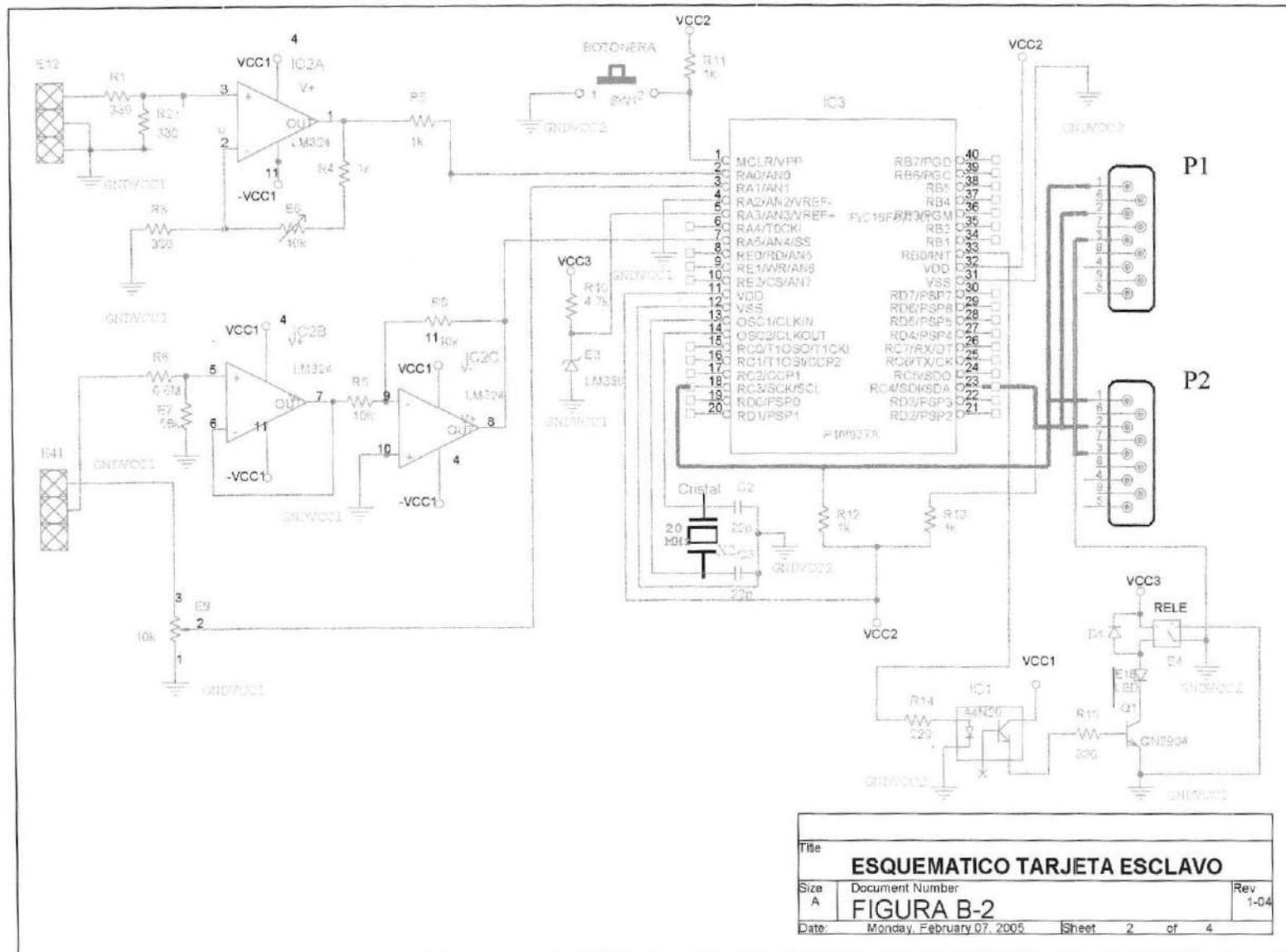
TABLA A-8

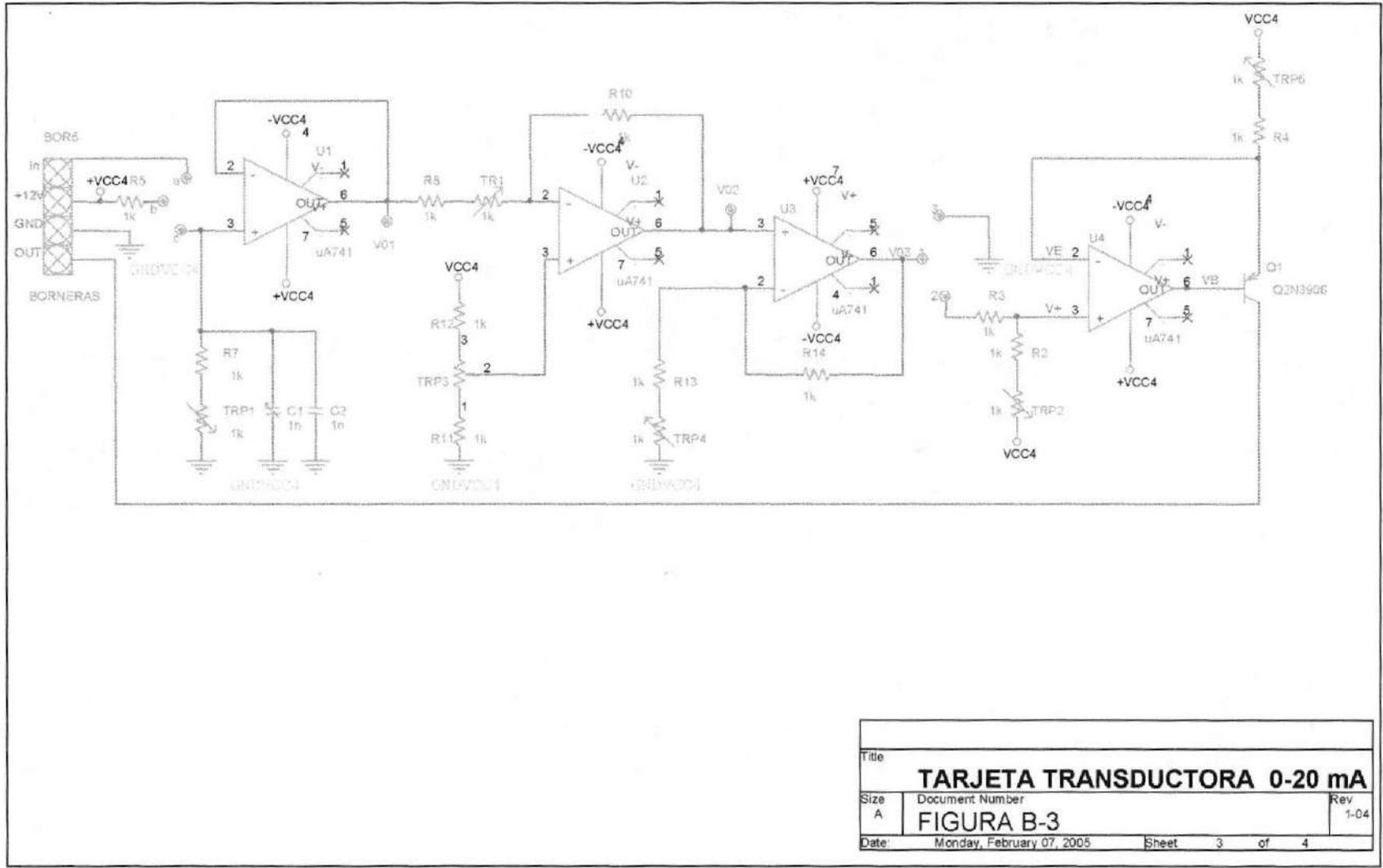
CARACTERISTICAS ELECTRICAS DEL  $\mu$ A7812 PARA  $V_I = 19V$ ,  $I_O = 500mA$ .

$\mu$ A7812						
Características	$T_J$	Min.	Tip.	Max.	Unid	Condición
Salida de Voltaje	25°C	11.5	12	12.5	V	IO = 5 mA a 1A, VI = 14.5 a 27V, PD $\leq$ 15 W
	0 a125°C	11.4	—	12.6		
Regulación de entrada de Voltaje	25°C	—	10	240	mV	VI = 14.5 V a 30 V
		—	3	120		VI = 16 V a 22 V
Rizado Rechazado	0 a125°C	55	71	—	dB	VI = 15 V a 25 V, f = 120 Hz
Regulación de salida de Voltaje	25°C	—	12	240	mV	IO = 5 mA a 1.5 A
		—	4	120		IO = 250 mA a 750 mA
Resistencia Salida	0 a125°C	—	0.018	—	$\Omega$	f = 1 kHz
Coefficiente de Temperatura de la salida de Voltaje	0 a125°C	—	-1	—	mV/ °C	IO = 5 mA
Ruido de Salida	25°C	—	75	—	$\mu$ V	f = 10 Hz a 100 kHz
Voltaje Dropout	25°C	—	2	—	V	IO = 1 A
Corriente Bias	25°C	—	4.3	8	mA	
Cambio de corriente Bias	0 a125°C	—	—	1	mA	VI = 14.5 V a 30 V
		—	—	0.5		IO = 5 mA a 1 A
Corriente Corto circuito de salida	25°C	—	350	—	mA	
Pico de Corriente de salida	25°C		2.2	—	A	

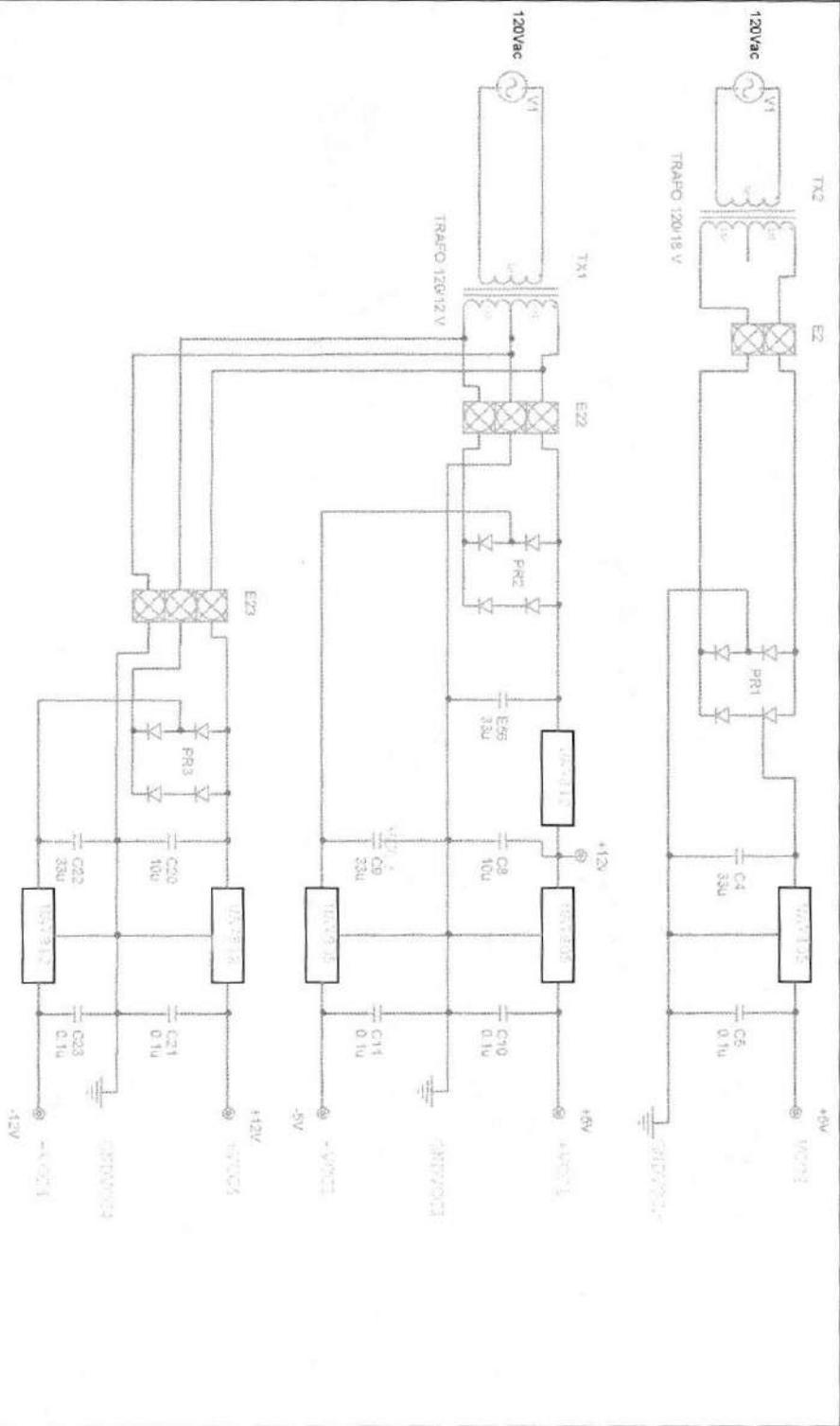
**B. DIAGRAMAS, ESQUEMATICOS DE TARJETAS ESCLAVO Y MAESTRO.**







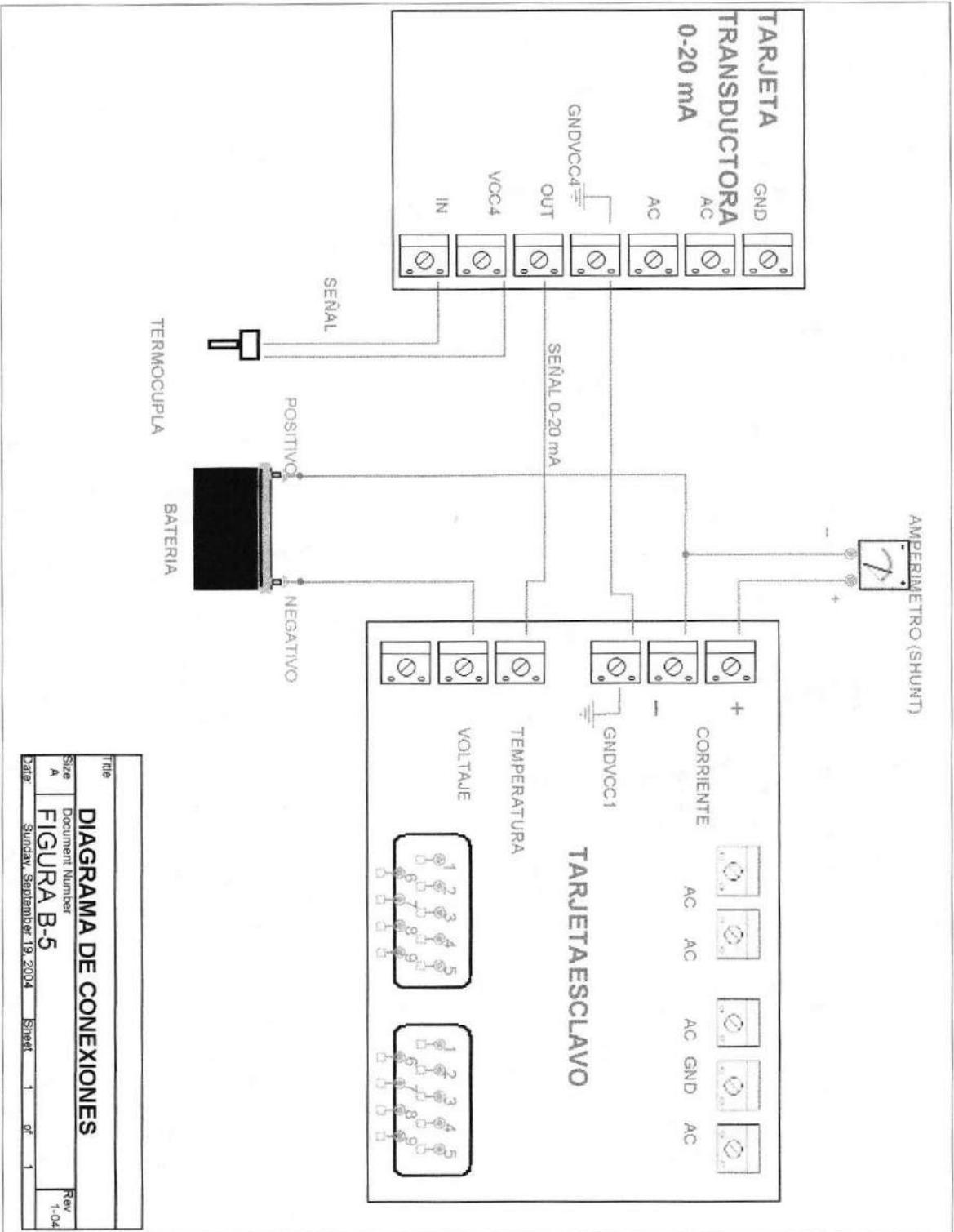
Title		
<b>TARJETA TRANSDUCTORA 0-20 mA</b>		
Size	Document Number	Rev
A	<b>FIGURA B-3</b>	1-04
Date:	Monday, February 07, 2005	Sheet 3 of 4



**ESQUEMATICO FUENTES DE VOLTAJE**

Size A  
 Document Number **FIGURA B-4**  
 Rev 1.04

Date: Monday, February 07, 2005 Sheet 4 of 4



**DIAGRAMA DE CONEXIONES**

Title	Document Number	Rev
Size	FIGURA B-5	1-04
Date	Sunday, September 19, 2004	Sheet 1 of 1



C. DIAGRAMAS DE CONEXIONES DE LOS MODULOS MAESTRO-  
ESCLAVOS Y CIRCUITOS IMPRESOS.

FIGURA C-1

CIRCUITO IMPRESO TARJETA ESCLAVO.

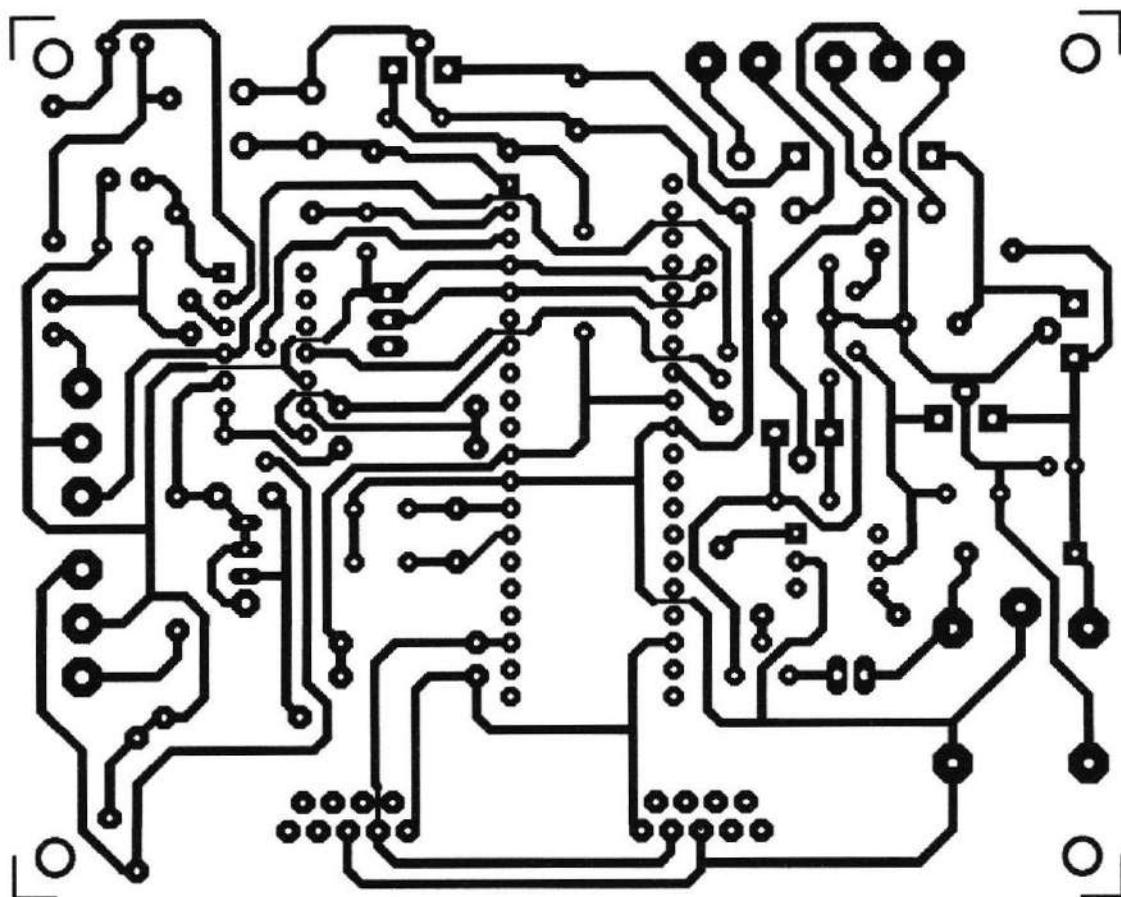


FIGURA C-2

DIAGRAMA DE POSICIONES TARJETA ESCLAVO

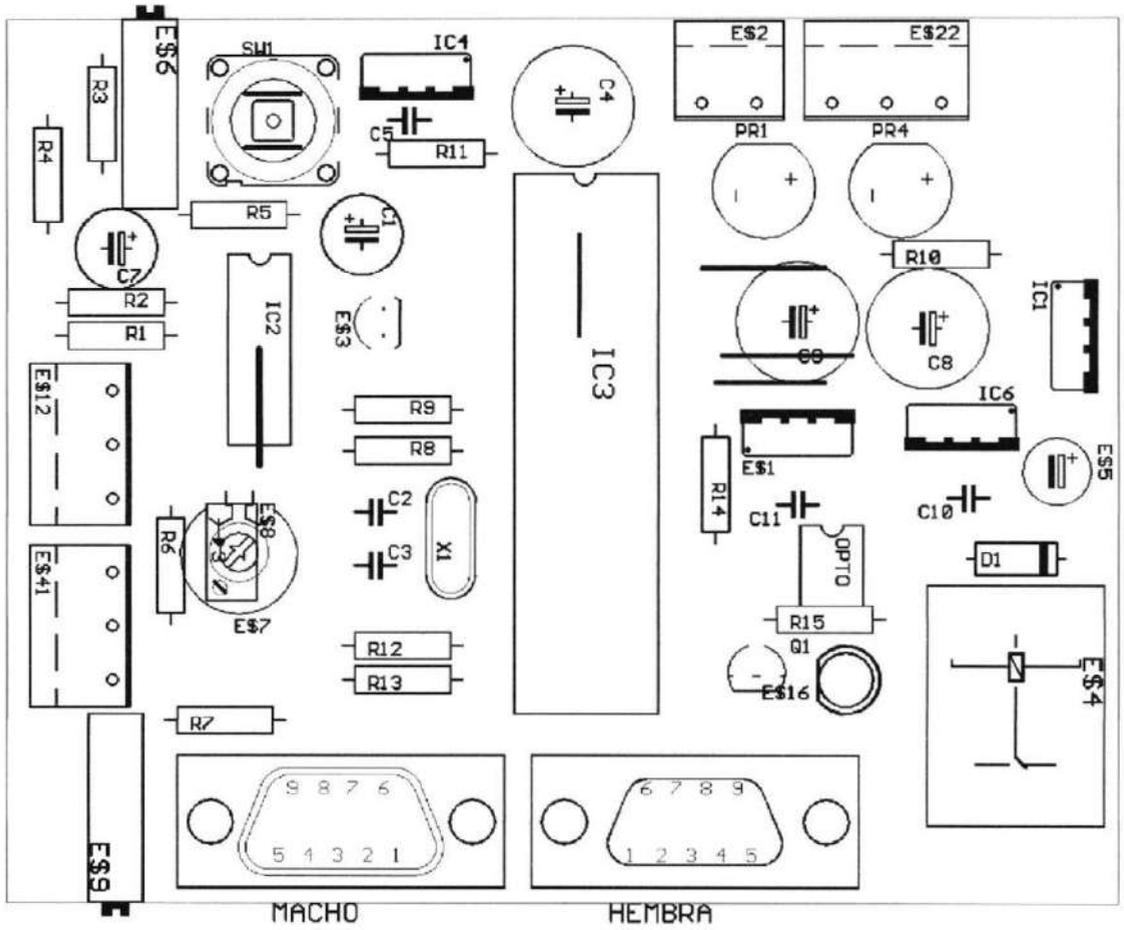


FIGURA C-3

CIRCUITO IMPRESO TARJETA MAESTRO

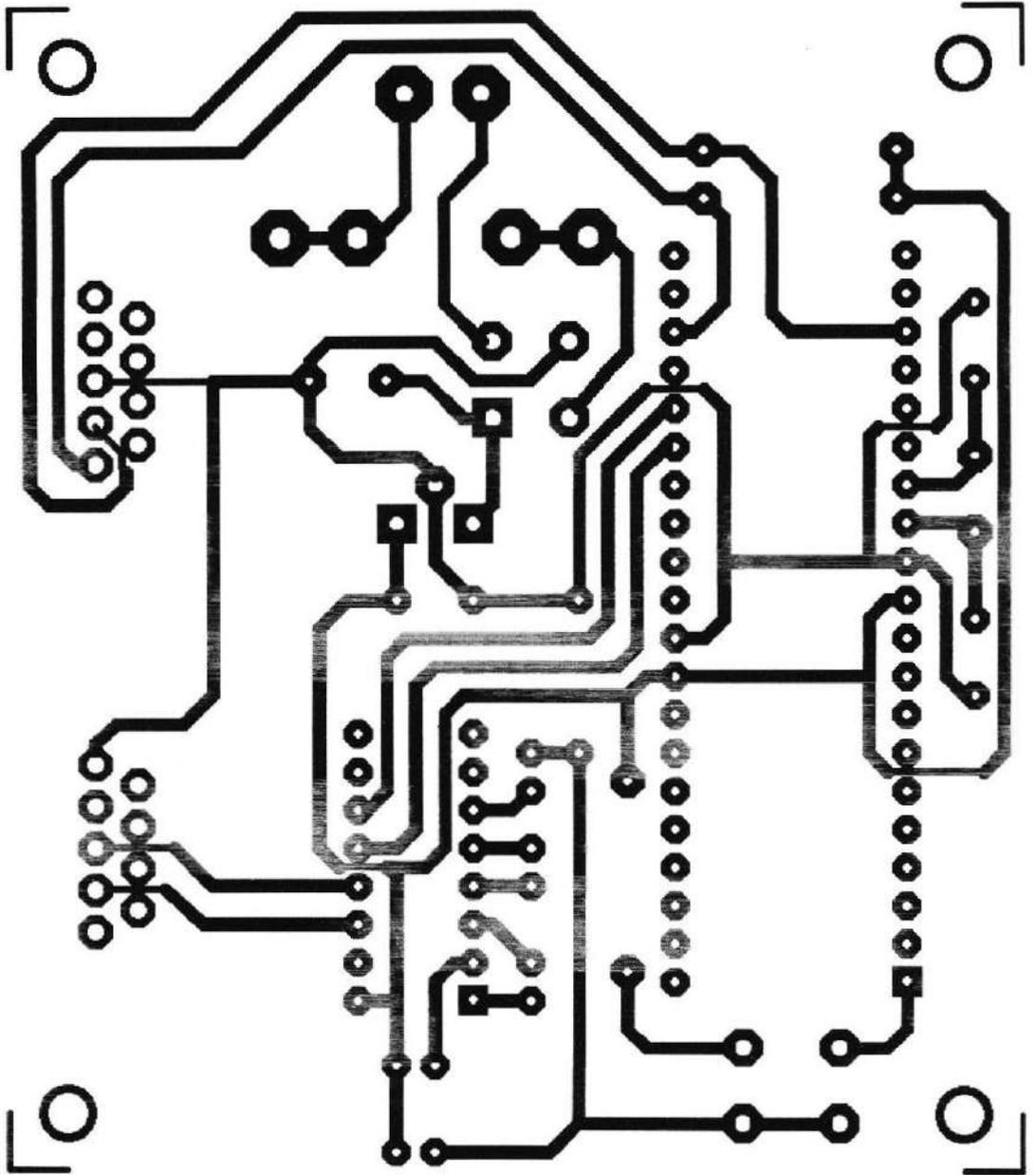
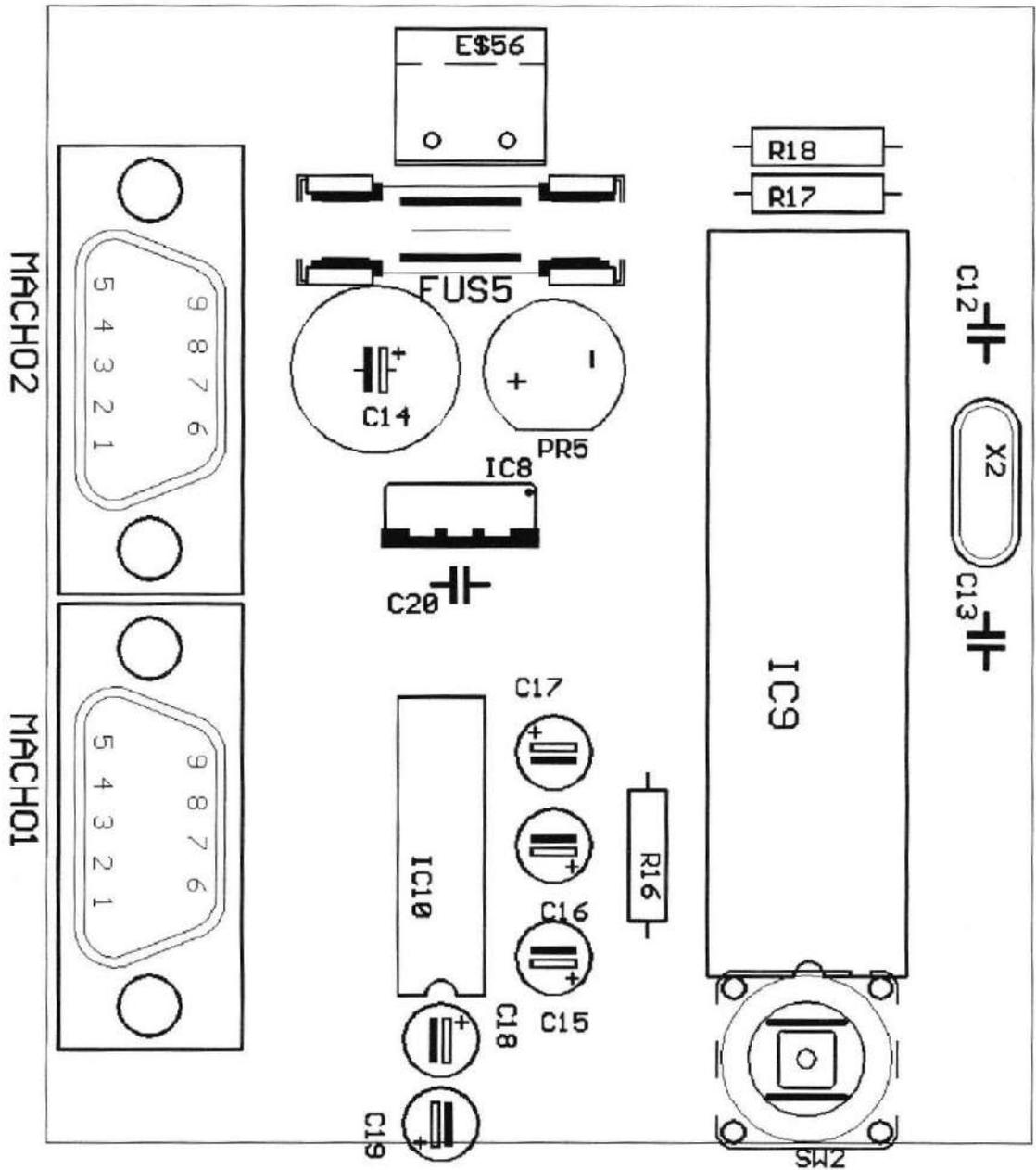


FIGURA C-4

DIAGRAMA DE POSICIONES TARJETA MAESTRO



**D. CODIGO FUENTE DE LOS MODULOS ESCLAVO, MAESTRO Y PUESTA EN MARCHA.**

*Código Fuente del Módulo Maestro*

Recibe datos de los esclavos. Funciona como Maestro y envía los datos a la PC. Transmite paquetes de datos por esclavo. Además este Firmware podría recibir hasta 368 datos por esclavo si esa fuese su aplicación pero en el caso del proyecto sólo recibimos 3 datos por esclavo.

\*\*\*\*\*

\* Programa para el PIC 18F877a recibe datos de los esclavos.

\*

\*\*\*\*\*

list P=16f877

Radix Hex

include <P16F877.inc>

```
Number_Data    EQU 0x20
Number_Esclavo EQU 0x21
Dir_Slave      EQU 0x22
Direccion      EQU 0x23
```

```
org    0x000
goto   Inicio
```

\*\*\*\*\*

### Inicializar Variables

\*\*\*\*\*

#### Ini\_Variables

```
    clwdt
    movlw 0x24           ; Localidad de dirección guarda los
                        ; datos
    movwf FSR           ; Del esclavo.
    clrf  Number_Data   ; Encera todas las variables.
    clrf  Number_Esclavo
    clrf  Dir_Slave
    clrf  Direccion
    return
```



CIB-ESPOL

\*\*\*\*\*

Inicializar Puertos.

\*\*\*\*\*

Ini\_Puertos

```
Clrwdt  
clrf PORTB  
clrf PORTC  
clrf PORTD  
clrf PORTE  
return
```

\*\*\*\*\*

.....  
Configuración puertos.  
.....

Config\_Port

Clrwdt

banksel TRISA ; Banco1

movlw b'00000110'

movwf ADCON1

clrf TRISA

clrf TRISB

movlw b'10111111'

movwf TRISC ; Puerto C configurado para USART.

clrf TRISD

clrf TRISE

banksel PORTA ; Banco0.

Return

.....

---

Config\_UsartRecib

```
Clrwdt
bsf  STATUS,RP0    ; Banco1
bcf  STATUS,RP1
movlw .129
movwf SPBRG       ; Velocidad de 9600 BADUD.
bcf  TXSTA,SYNC   ; Modo asincrónico.
bsf  TXSTA,BRGH   ; Alta velocidad y 0.16 % de error.
bcf  STATUS,RP0   ; Banco 0.
bsf  RCSTA,SPEN   ; Habilitado el Usart.
bsf  RCSTA,CREN   ; Habilitada la recepción da la orden
return            ;De la PC lo hace con recibimiento
                ;continuo.
```

\*\*\*\*\*

Se habilita el Usart del Pic para recibir la orden de Atención de la PC para recibir los datos de las tarjetas. Cada cierto tiempo, este tiempo de pedido de datos está programado en el software de la PC.

\*\*\*\*\*

Recep\_PC

```
Clrwdt
Banksel RCSTA      ; Banco0.
movlw b'00000001'
movwf PORTE
Clrwdt
btfss  PIR1,RCIF   ; Espera recibir llamada de atención
                               ;de PC.
goto   $-1         ; Espera.
movf   RCREG,W     ; Se lee el registro RCREG.
movwf  PORTB
sublw  b'1100110'  ; Comprueba llamada de atención
                               ; de la PC
btfss  STATUS,Z    ; Es la correcta.
goto   Recep_PC   ; Es la llamada correcta?
```

```

movlw b'00000111' ; No. espere otra llamada.
movwf PORTE
bcf RCSTA,CREN ; Deshabilita el USART como receptor.
bcf RCSTA,SPEN ; Deshabilita el USART.
return

```

\*\*\*\*\*

Velocidad 100 KHZ clock=FOSC/(4\*(SSPADD+1)).

\*\*\*\*\*

Velocidad

```

Clrwdt
Banksel TRISC
movlw 0xFF
movwf TRISC ; Puerto C como entrada para I2c.
movlw .255
banksel SSPADD ; Banco1.
movwf SSPADD
movlw b'100000000' ; SMP=1 y CKE=0
movwf SSPSTAT
bcf STATUS,RP0 ; Banco0.
return

```

\*\*\*\*\*

Habilitado módulo I2c (SSPEN, y modo maestro).

\*\*\*\*\*

Config\_I2c

```
Clrwdt
banksel PORTA
clrf PORTA
clrf PORTB
clrf PORTD
movlw b'00101000'
movwf SSPCON
return
```

.....

Envía condición de inicio.

\*\*\*\*\*

Envi\_Start

```
Clrwdt
banksel SSPCON2 ; Banco1.
```

```
bsf    SSPCON2,SEN ; Start se inicia.  
Clrwdt  
btfsc  SSPCON2,SEN  
goto   $-1  
Clrwdt  
banksel PIR1 ; Banco0.  
btfss  PIR1,SSPIF  
goto   $-1  
movlw  b'00000001'  
movwf  PORTE  
movwf  PORTE  
return
```

\*\*\*\*\*

Send\_Dir

```
Clrwdt  
bcf    PIR1,SSPIF  
movf   Dir_Slave,W  
movwf  SSPBUF  
Clrwdt  
btfss  PIR1,SSPIF
```

```
goto    $-1
movlw   b'00000010'
movwf   PORTE
movwf   PORTA
call    SPCON2B
movwf   PORTB
return
```

\*\*\*\*\*

#### Recib\_Dato

```
Clrwdt
bcf     PIR1,SSPIF
banksel SSPCON2      ; Banco1
bsf     SSPCON2,RCEN
Clrwdt
btfsc  SSPCON2,RCEN
goto    $-1
Clrwdt
Banksel PIR1          ; Banco0.
btfss  PIR1,SSPIF
goto    $-1
```

```

movlw  b'00000111'
movwf  PORTE
movwf  PORTA
call   SPCON2B
movwf  PORTB
return

```

\*\*\*\*\*

ACK

```

bcf    PIR1,SSPIF           ; Reponer flag.
banksel SSPCON2            ; Banco1.
bcf    SSPCON2,ACKDT       ; Generar ACK maestro.
bsf    SSPCON2,ACKEN       ; Se genera el ACK.
Clrwdt
btfsc  SSPCON2,ACKEN
goto   $-1
Clrwdt
bcf    STATUS,RP0          ; Banco0
btfss  PIR1,SSPIF         ; Finalizó la cond. ACK?.
goto   $-1                 ; No. Esperar.
movf   SSPBUF,W           ; Dato del esclavo en W.

```



```
movwf PORTB
movwf INDF
movlw b'00000100'
movwf PORTE
movwf PORTA
return
```

\*\*\*\*\*

#### NACK

```
Clrwdt
bcf    PIR1,SSPIF           ; Reponer flag.
banksel SSPCON2           ; Banco1.
bsf    SSPCON2,ACKDT       ; Generar ACK maestro.
bsf    SSPCON2,ACKEN       ; Se genera el ACK.
Clrwdt
btfsc  SSPCON2,ACKEN
goto   $-1
clrwdt
bcf    STATUS,RP0          ; Banco0
btfss  PIR1,SSPIF         ; Finalizó la cond. ACK?.
goto   $-1                 ; No. Esperar.
```

```
movf   SSPBUF,W           ; Dato del esclavo en W.
movwf  PORTB
movwf  INDF
movlw  b'00000101'
movwf  PORTE
movwf  PORTA
return
```

\*\*\*\*\*

STOP

```
Clrwdt
bcf    PIR1,SSPIF        ; Reponer flag.
banksel SSPCON2         ; Banco1.
bsf    SSPCON2,PEN
clrwdt
btfsc  SSPCON2,PEN
goto   $-1
Clrwdt
bcf    STATUS,RP0       ; Banco0
btfss  PIR1,SSPIF
goto   $-1
```

```
movlw b'00000110'  
movwf PORTE  
return
```

\*\*\*\*\*

### Config\_UsartTrans

```
Clrwdt  
  
bcf  SSPCON,SSPEN ; Deshabilita el I2C.  
bsf  STATUS,RP0   ; Banco1  
bcf  STATUS,RP1  
  
movlw b'10111111'  
  
movwf TRISC       ; Puerto C configurado para Usart.  
movlw .129        ; Velocidad = Fosc / (64 (X + 1))  
movwf SPBRG       ; Velocidad de 9600 BAUD.  
bcf  TXSTA,SYNC   ; Modo asincrónico.  
bsf  TXSTA,BRGH   ; Alta velocidad y 0.16 % de error.  
bsf  TXSTA,TXEN   ; Habilita el USART como transmisor.  
bcf  STATUS,RP0   ; Banco 0.  
bsf  RCSTA,SPEN   ; Habilitado el USART.  
  
Return
```

\*\*\*\*\*

## Memory\_Control

```

        clrwdt
        btfsc STATUS,IRP
        goto Banco2_3
Banco0_1 Clrwdt
        movlw 0x80          ; Lazo donde IRP=0 es decir esta en
banco0.
        subwf FSR,W
        btfss STATUS,Z      ; Está en el límite del banco 0?
        goto banco_1       ; No. Conserva localidad de memoria
                            ; Anterior en banco0.
        movlw 0xA0         ; Si. Va a localidad de memoria 0xA0
                            ; En Banco1.
        movwf FSR          ; FSR en banco1.
        bcf STATUS,IRP     ; IRP = 0.
        goto Regresa      ; Regresa.
banco_1 Clrwdt
```

```

movlw 0xF1          ; IRP todavía en 0 es decir banco1.
subwf FSR,W
btfss STATUS,Z     ; Esta en el limite del banco1?
goto Regresa       ; No. Regresa y conserva la localidad
                    ; De memoria anterior.

movlw 0x10          ; Si. Cambia a banco2 con localidad en
                    ; 0x20.

movwf FSR           ; FSR=0x10.
bsf STATUS,IRP     ; IRP=1.
goto Regresa       ; Regresa.

```

Banco2\_3

```
Clrwdt
```

```
movlw 0x71          ; Lazo donde IRP=1 es decir esta en
```

banco2.

```
subwf FSR,W
```

```
btfss STATUS,Z     ; Está en el límite del banco 2?
```

```
goto banco_3       ; No. Conserva localidad de memoria
                    ; Anterior en banco2.
```

```
movlw 0x90          ; Si. Va a localidad memoria 0x90 en
```

Banco3.

movwf FSR ; FSR en banco3.

bsf STATUS,IRP ; IRP = 1.

goto Regresa ; Regresa.

banco\_3 Clrwdt

movlw 0xF1 ; IRP todavía en 1.

subwf FSR,W

btfss STATUS,Z ; Esta en el limite del banco3?

goto Regresa ; No. Regresa y conserva localidad de

; Memoria anterior.

; Código cuando se agote la memoria

;RAM.

goto Regresa ; Regresa.

Regresa

return

.....

Buffer\_Salida

Clrwdt

banksel TXREG ; Carga el siguiente dato en reg.

movwf TXREG ; De salida TXREG.

```

Clrwdt
btfss   PIR1,TXIF      ; Se desalojo el registro de salida
                               ;TXREG.
goto    $-1           ; No. Esperar.
banksel TXSTA          ; Banco 1.
Clrwdt
btfss   TXSTA,TRMT    ; Se transmitieron el dato?.
goto    $-1           ; No. espere.
bcf     STATUS,RP0    ; Banco 0.
bcf     STATUS,RP1
return

```

\*\*\*\*\*

Se transmite a la PC los datos de cada esclavo. Puede transmitir hasta 365 datos por esclavo y los datos se guardan en la RAM mientras que la Dirección de los esclavos que respondieron y el número de Datos que soporta en la EEPROM de datos.

\*\*\*\*\*

Transmitir

```

clrwdt
movlw  0x24

```

```

movwf FSR           ; Se saca el número de
bcf STATUS,IRP     ; Datos del esclavo.
movf INDF,W        ; Y se lo guarda en
movwf Number_Data  ; Number_Data.
incf Direccion,f   ; Se va a la siguiente dirección.
call Guarda_EPROM  ; Guarda en memoria el número de
                    ; datos
movlw 0x24         ; De todos los Esclavos.
movwf FSR         ; Se guarda la dirección del
bcf STATUS,IRP    ; Esclavo en la memoria RAM.
movf Dir_Slave,W  ; En pos. 0x24 del banco0.
movwf INDF
incf Direccion,f  ; Se incrementa la
call Guarda_EPROM ; Dirección y se lo guarda en la
movlw b'00000111' ; EEPROM de Datos.
movwf PORTE
movlw b'001000'
movwf PORTA
temp
clrwdt
incf FSR,f

```

```

call    Memory_Control
movf    INDF,W
call    Buffer_Salida
decfsz Number_Data,f
goto    temp
movlw   b'00000011'    ; Se transmitieron los datos a la PC.
movwf   PORTE
banksel TXSTA          ; Banco1.
bcf     TXSTA, TXEN    ; Se deshabilita el Usart como
                        ;transmisor.
bcf     STATUS, RP0    ; Banco0.
bcf     RCSTA, SPEN
return

```

\*\*\*\*\*

Guarda\_EPROM

```

Clrwdt
bsf     STATUS, RP1    ; Banco3
bsf     STATUS, RP0
Clrwdt

```

```

btfsc  EECON1, WR    ; Verifica si escritura está en proceso:
goto   $-1           ; Si. Espera a que termine la escritura.
Banksel Direccion    ; Banco0.
movf   Direccion,W   ; Se guarda la dirección
Banksel EEADR        ; Banco2
movwf  EEADR         ; En EEADR.
bcf    STATUS,IRP
movlw  0x24
movwf  FSR
movf   INDF,W        ; Guardo el valor en
movwf  EEDATA        ; EEDATA.
bsf    STATUS,RP0    ; Banco3.
bcf    EECON1,EEPGD ; Escoge escritura en data memory.
bsf    EECON1,WREN   ; Habilita la escritura.
movlw  0x55          ; Pasos necesarios.
movwf  EECON2        ; Para la transmisión
movlw  0xAA
movwf  EECON2
bsf    EECON1,WR     ; Escritura en proceso.
bcf    STATUS,RP0    ; Banco0.

```

```

bcf     STATUS,RP1

Clrwdt

btfss  PIR2,EEIF      ; Terminó de escribir?

goto   $-1            ; No. Espere

bcf     PIR2,EEIF     ; Si. Repone flag

banksel EECON1        ; Banco3.

BCF     EECON1, WREN

bcf     STATUS,RP0    ; Banco0.

bcf     STATUS,RP1

return

```

\*\*\*\*\*

### Lee\_EPROM

```

Clrwdt

bsf     STATUS, RP1

bcf     STATUS, RP0   ; Banco2

banksel Direccion    ; Banco0.

movf    Direccion, W  ; Escribe Dirección.

banksel EEADR        ; Banco2.

movwf   EEADR        ; Lee de la dirección especificada.

bsf     STATUS, RP0   ; Banco3

```



```

bcf    EECON1, EEPGD ; Puntero a la memoria del dato.
bsf    EECON1, RD    ; Empieza la operación de Lectura.
bcf    STATUS, RP0   ; Banco2
movf   EEDATA, W     ; W = EEDATA
bcf    STATUS, RP0   ; Banco0.
bcf    STATUS, RP1
return

```

\*\*\*\*\*

Inicio

```

call   Ini_Variables ; Inicializa las variables usadas.
call   Ini_Puertos   ; Limpia los puertos usados.

```

Next\_Dir

```

Clrwdt
movlw  b'00000011'
movwf  Dir_Slave
call   Config_Port   ; Configura puertos como entradas o
                    ;salidas.
call   Config_UsartRecib ; Configura modo USART para recibir
                    ; Llamada de PC.
call   Recep_PC      ; Recibe llamada de atención de la PC.

```

Next

```
Clrwdt  
  
bcf    STATUS,IRP  
  
movlw  0x24  
  
movwf  FSR  
  
call   Velocidad      ; Configura la velocidad en I2C.  
call   Config_I2c     ; Configura el Pic en modo I2C.  
call   Envi_Start     ; Envía la condición de START.  
call   Send_Dir       ; Envía la dirección del esclavo.  
  
Banksel SSPCON2      ; Banco1.  
  
btfsc  SSPCON2,ACKSTAT  
  
goto   Abort_Transm  
  
bcf    STATUS,RP0    ; Banco0.  
  
incf   Number_Esclavo,f  
  
call   Recib_Dato     ; Recibe el dato del esclavo.  
call   ACK            ; Recibe condición ACK del esclavo.  
movf   INDF,W        ; Recoge de memoria número de  
                                ; Datos enviado por el esclavo  
  
movwf  Number_Data   ; Y lo guarda en Number_Data.
```

Next\_Dato

Clrwdt

incf FSR,f ; Incrementa localidad de memoria.

call Memory\_Control

call Recib\_Dato ; Recibe el siguiente dato.

decfsz Number\_Data,f ; Recibió todo los datos?

goto Manda\_ACK ; No. Manda otra cond. ACK

call NACK ; Si. Manda cond. NACK.

call STOP ; Envía condición de STOP. Fin de la  
; Transmisión en I2C.

call Config\_UsartTrans

call Transmitir

movlw b'00000010' ; Asigna dirección siguiente esclavo.

addwf Dir\_Slave,f ; Con esta constante indirectamente  
; Tengo el número

; A este punto llega así sea que

; Responda o no el esclavo.

End\_Dir

Clrwdt

```

movlw  b'11111111'    ; Averigua si se termino de
                        ; Direccionar los esclavos.
subwf  Dir_Slave,W    ;Se pueden direccionar 128
                        ;esclavos.
btfss  STATUS,Z      ; Terminó de Direccionar?
goto   Next          ; No. Siga siguiente esclavo
movlw  b'11111111'
andwf  Number_Esclavo,W
                        ; Si no hay esclavos conectados
                        ; no Envía a la PC.
btfss  STATUS,Z      ; Envía a la PC?
goto   Manda_Dir    ; Si. Envíe las direcciones de los
                        ; Esclavos que respondieron.
incf   Direccion,f
goto   Manda_Dir    ; No. Espere otra llamada.

```

Manda\_ACK

```

Clrwdt
call   ACK          ; Envía cond. ACK al siguiente dato.
goto  Next_Dato    ; Recibe siguiente dato.

```

Abort\_Transm

Clrwdt

bcf STATUS,RP0 ; Banco0.

call STOP

bcf SSPCON,SSPEN ; Deshabilita el I2C.

movlw b'00000010' ;Se asigna la dirección del siguiente  
;esclavo.

addwf Dir\_Slave,f ;Con esta constante indirectamente  
;tengo el Número

goto End\_Dir

Manda\_Dir

Clrwdt

call Config\_UsartTrans

lazo

clrwdt

call Lee\_EPROM

call Buffer\_Salida

decfsz Direccion,f

goto lazo

movf Number\_Esclavo,W

```

call    Buffer_Salida
banksel TXSTA      ; Banco1.
bcf     TXSTA, TXEN ; Se deshabilita el USART como
                        ; transmisor.
bcf     STATUS, RP0 ; Banco0.
bcf     RCSTA, SPEN
goto    Inicio
END      ; Fin.

```

*Código Fuente del Módulo Esclavo*

El módulo esclavo transmite al módulo maestro 3 variables. El Firmware realiza el promedio de las lecturas de los datos cada cierto tiempo el cual se establece en software "Sistema de Monitoreo".

\*\*\*\*\*

\*Programa para el PIC 18F877 en modo esclavo, comunicación

I2C.\*

\*\*\*\*\*

list p=16f877a

Radix Hex

include <P16F877.inc>

Delay	EQU	0x20
DatoL	EQU	0x21
DatoH	EQU	0x22
Number_Datos	EQU	0x23
Cociente_L	EQU	0x24
Cociente_H	EQU	0x25
Divisor_L	EQU	0x26
Divisor_H	EQU	0x27
MultiA	EQU	0x28
MultiB	EQU	0x29
DatoA_H	EQU	0x2A
DatoA_L	EQU	0x2B
DatoB_H	EQU	0x2C
DatoB_L	EQU	0x32

DatoD_H	EQU 0x2D
DatoD_L	EQU 0x2E
DatoC_H	EQU 0x2F
DatoC_L	EQU 0x30
MultiC	EQU 0x31
CONTADOR	EQU 0x33
Acumula_L	EQU 0x34
Acumula_H	EQU 0x35
Dato_32H	EQU 0x36
DatoB_32H	EQU 0x37
vuelta	EQU 0x38
Tiempo	EQU 0x39
Direccion	EQU 0x3A
Dato	EQU 0x3B

org 0x000

goto inicio

org 0x005

\*\*\*\*\*

Muestra el status de SSPSTAT en el puerto D.

\*\*\*\*\*

SPSTATD

Clrwdt

banksel SSPSTAT ; Banco1.

movf SSPSTAT,W

bcf STATUS,RP0 ; Banco0.

movwf PORTD

return

\*\*\*\*\*

\*Muestra SSPCON en PuertoB.\*

\*\*\*\*\*

SPCONB

Clrwdt

movf SSPCON,W

movwf PORTB

return

\*\*\*\*\*

\*Muestra el status de SSPCON2 en el puerto B.\*

\*\*\*\*\*

SPCON2B

```
Clrwdt
banksel SSPCON2      ; Banco1.
movf  SSPCON2,W
bcf   STATUS,RP0     ; Banco0.
movwf PORTB
return
```

\*\*\*\*\*

\*Muestra el status de PIR1 en el puerto B.\*

\*\*\*\*\*

SPIR1B

```
Clrwdt
banksel PIR1         ; Banco0.
movf  PIR1,W
movwf PORTB
return
```

\*\*\*\*\*

\* Muestra el status de PIR2 en el puerto A.\*

\*\*\*\*\*

PIR2A

```
Clrwdt
movf   PIR2,W
movwf  PORTA
return
```

\*\*\*\*\*

\* Limpia los puertos.\*

\*\*\*\*\*

Ini\_Puertos

```
Clrwdt
clrf   PORTA
clrf   PORTB
clrf   PORTC
clrf   PORTD
clrf   PORTE
clrf   Cociente_L
clrf   Cociente_H
```

clrf	Divisor_L
clrf	Divisor_H
clrf	DatoA_L
clrf	DatoB_H
clrf	DatoB_L
clrf	DatoD_H
clrf	DatoD_L
clrf	DatoC_H
clrf	DatoC_L
clrf	MultiC
clrf	CONTADOR
clrf	Acumula_L
clrf	Acumula_H
clrf	DatoL
clrf	DatoH
clrf	Dato_32H
clrf	DatoB_32H
clrf	SSPCON
clrf	Tiempo
clrf	Direccion

```
clrf    Dato
clrf    vuelta
return
```

\*\*\*\*\*

### Config\_Puertos

```
Clrwdt

banksel TRISC          ; Banco1
movlw  b'10011110'    ; W=0xFF
movwf  TRISC          ; PuertoC entrada I2C,RC0,RC5,RC6
                        ;salida.

movlw  b'00000110'    ; Entradas o Salidas como digitales.
movwf  ADCON1         ; Van en ADCON1.

movlw  b'11111111'    ; W=0xFF
movwf  TRISA          ; PuertoA como Entrada.

clrf   TRISB          ; PuertoB como salida para pruebas.

clrf   TRISD          ; PuertoD como salida.

movlw  b'11111111'    ; W=0xFF
movwf  TRISE          ; PuertoE como Entrada.

bcf   STATUS,RP0     ; Banco0.

return
```

\*\*\*\*\*

Config\_I2C

```
Clrwdt  
  
Banksel SSPADD ; Banco1  
  
; En SSPADD va la dirección del  
  
; Esclavo.  
  
movlw b'00000111' ; W = 0x0A  
movwf SSPADD ; Guarda dirección del esclavo en  
  
; SSPADD.  
  
bsf SSPSTAT,SMP ; Velocidad estándar de 100 KHZ.  
banksel SSPCON ; Banco 0  
movlw b'00110110' ; SSPEN = 1 Habilita USART, CKP=1  
  
; Clock habilitado.  
  
movwf SSPCON ; Modo Esclavo en I2C.  
  
return
```

\*\*\*\*\*

Interrupt\_on

```
Clrwdt  
  
banksel PIE1 ; Banco1.
```



```

bsf    PIE1,SSPIE    ; Habilita interrupciones I2C para
                        ; Despertar del sleep.

movlw  b'01000000'  ; Habilitada las interrupciones GIE=0
movwf  INTCON        ; Y PEIE=1.
bcf    STATUS,RP0    ; Banco0.

return

```

\*\*\*\*\*

#### Conv\_AD

```

bcf    PIR1,ADIF      ; repone Flag del convertidor.

clrwdt

bsf    ADCON0,GO      ; inicia la conversión
btfss  PIR1,ADIF      ; Terminó la conversión?
goto   $-1            ; No. espere.

Banksel ADRESL        ; Banco1
movf   ADRESL,W        ; Se lee el dato convertido.
bcf    STATUS,RP0     ; Banco0.

Movwf  DatoL

movf   ADRESH,W        ; Se lee el dato convertido.

movwf  DatoH

return

```

\*\*\*\*\*

## Timer

; Lazo de temporización.

Movwf Delay ; Se guarda en la variable Delay.

### timer\_1

Clrwdt

movlw .240 ; Valor en TMR0 para temporización.

movwf TMR0

bcf INTCON,T0IF ; Repone flag de desbordamiento del  
;TMR0.

### timer\_2

Clrwdt

btfss INTCON,T0IF ; Se desbordó el timer?

goto timer\_2 ; No. Espere.

Decfsz Delay,f ; Termino lazo de temporización ?

goto timer\_1 ; No. Espere.

return

\*\*\*\*\*

Division

Clrwdt

clrf Cociente\_L

clrf Cociente\_H

goto jump2+0D

jump2

movf Acumula\_L,W ; El Dato\_L/residuo a W.

movwf DatoB\_L ; El Dato\_L/residuo va a ser  
;multiplicado  
; Para sucesivas

movf Acumula\_H,W ; Divisiones en DatoB.

movwf DatoB\_H ; Parte alta de DatoB.

movf Dato\_32H,W

movwf DatoB\_32H

movlw .1 ; Cociente = 1.

addwf Cociente\_L,1 ; Aumenta cociente.

btfss STATUS,C ; Se desborda Cociente\_L?

goto \$+2 ; No. Salta.

```

incf    Cociente_H,1    ; Si. Sume el acarreo C<STATUS> al
                               ; Cociente_H.
bsf     STATUS,C        ; C acarreo =1.
movf    Divisor_L,W     ; Divisor.
subwfm Acumula_L,1     ;"Restas sucesivas" Resta Dividendo y
                               ; Divisor.
btfsc   STATUS,C        ; Acarreo?
goto    jumpex          ; No. Vaya a jump.
incf    Divisor_H,W     ;
goto    jumpex+1

```

jumpex

```

movf    Divisor_H,W     ; Parte alta del divisor.
subwfm Acumula_H,1     ; Se resta parte alta del divisor.
btfsc   STATUS,C        ;El Dividendo < Divisor.
goto    jump2
movlw   .1
subwfm Dato_32H
btfsc   STATUS,C
goto    jump2
return

```

\*\*\*\*\*

## Multiplicacion

```
Clrwdt
movlw .16
movwf CONTADOR
clrf DatoD_H
clrf DatoD_L
```

## Loop

```
rrf DatoB_H,1
rrf DatoB_L,1
btfss STATUS,C
goto No_Suma
movf DatoA_L,W
addwf DatoD_L,1
btfsc STATUS,C
incf DatoD_H,1
movf DatoA_H,W
addwf DatoD_H,1
```

No\_Suma

```
clrwdt  
rrf    DatoD_H,1  
rrf    DatoD_L,1  
rrf    DatoC_H,1  
rrf    DatoC_L,1  
decfsz CONTADOR,1  
goto   Loop  
return
```

\*\*\*\*\*

Buffer\_Salida

```
Clrwdt  
bcf    SSPCON,WCOL ; Limpia el bit WCOL.  
movwf  SSPBUF      ; Escribe el byte en WREG.  
bcf    SSPCON,WCOL ; Limpia el bit WCOL.  
bsf    SSPCON,CKP  ; Suelta el control del Reloj..  
Clrwdt  
btfss  PIR1,SSPIF ; Transmitió Dato?  
goto   $-1        ; No. Espere
```

```
bcf    PIR1,SSPIF      ; Reponer Bandera..
return
```

\*\*\*\*\*

\* Promedio de lecturas 255 lecturas por dato y luego se saca el promedio.\*

\*\*\*\*\*

### Promedio

```
Clrwdt
movlw  .255          ; Cantidad datos leídos en un canal
movwf  CONTADOR     ; el Promedio son 255.
movlw  .1
movwf  vuelta
clrf   Cociente_L   ; Limpio todas las variables.
clrf   Cociente_H
clrf   Divisor_L
clrf   Divisor_H
clrf   DatoA_L
clrf   DatoB_H
clrf   DatoB_L
clrf   DatoD_H
clrf   DatoD_L
```

```
clrf    DatoC_H
clrf    DatoC_L
clrf    MultiC
clrf    Acumula_L
clrf    Acumula_H
clrf    DatoL
clrf    DatoH
clrf    Dato_32H
clrf    DatoB_32H
```

jump

```
Clrwdt
movlw  .1
call   Timer           ; Tiempo de adquisición del canal del
                        ; A/D.
call   Conv_AD         ; Rutina para conversión de señal
                        ; analógica.
Clrwdt
movf   DatoL,W
addwf  Acumula_L,1    ; Se suman las lecturas la parte baja.
btfss  STATUS,C       ; Hay acarreo?
```

goto     \$+3                     ; No. Sume la parte alta de la lectura.  
 incf     DatoH,W               ; Si. Sume el acarreo.  
 goto     \$+2                     ; No hay acarreo vaya a sumar la parte  
                                    ; alta.  
 movf     DatoH,W               ; Suma la parte alta de la lectura.  
 addwf    Acumula\_H,1           ; Suma parte alta de los valores  
                                    ; acumulados.  
 btfss    STATUS,C             ; Hay acarreo?  
 goto     \$+2                     ; No. Sume la parte alta de la lectura.  
 incf     Dato\_32H,1           ; Si. Sume el acarreo.  
 decfsz   CONTADOR,1          ; Cuenta 63 lecturas.  
 goto     jump                  ; Vaya a tomar otro dato del mismo  
                                    ; canal.  
 decfsz   vuelta,1  
 goto     jump  
 clrwdt  
 movf     Acumula\_L,W          ; El Acumulado parte baja a W.  
 movwf    DatoB\_L               ; Acumulado parte baja multiplicado  
                                    ; Para Sucesivas.  
 movf     Acumula\_H,W          ; Divisiones en DatoB.



```

movwf DatoB_H      ; Parte alta DatoB está el residuo o el n
                   ; Número original.

movf Dato_32H,W    ; Para suma de números mayores a 16
                   ; bits.

movwf DatoB_32H    ; Capacidad para 32 bits.

bcf STATUS,C       ; Acarreo =1.

movlw 0xFF

movwf Divisor_L    ; Parte baja del divisor.

movlw 0x00

movwf Divisor_H    ; Parte alta del Divisor.
                   ; Comprueba si Dividendo > divisor.

movlw .0

iorwf Dato_32H,W   ; Comprueba número acumulado mayor
                   ; de 16 bits.

btfss STATUS,Z     ; Mayor a 16 bits?

goto Div           ; Si.Vaya Dividir.
                   ; Cualquier número >16 bits mayor a
                   ; 255.

movf Divisor_L,W

```

```

subwf  Acumula_L,0    ;"Restas sucesivas" Resta Dividendo y
                        ; Divisor (Parte Baja).

btfsc  STATUS,C      ; Acarreo?

goto   jumpln        ; No. Vaya a jumpln.

incf   Divisor_H,W   ; Si. Reste el acarreo.

Goto   jumpln+2;+1

```

jumpln

```

clrwdt

movf   Divisor_H,W   ;Parte alta del divisor.

subwf  Acumula_H,0   ;Se resta parte alta del divisor.

btfsc  STATUS,C      ;El dividendo < Divisor.

goto   Div           ;No.vaya a dividir.

clrf   Cociente_L    ;Cociente cero.

goto   Div+2;+1      ;Si.vaya a multiplicar el numero por 10.

```

Div

```

clrwdt

call   Division      ;llama a procedimiento Division.

movf   Cociente_L,W  ;El cociente a W.

movwf  MultiA

movf   Cociente_H,W  ; El cociente a W.

```

```

movwf MultiB
movlw .10 ; Multiplica DatoB por 10.
movwf DatoA_L ; En datoA va el multiplicando.
clrf DatoA_H

```

Multi

```

Clrwdt
call Multiplicacion ; Llama al Procedimiento
;multiplicación.
movf Divisor_L,W ; Divisor en W.
subwf DatoC_L,W ; (DatoC*10)-Divisor
btfsc STATUS,C ; Es menor (DatoC+10)?
goto jumpIn2 ; No vaya a restar la Parte alta.
incf Divisor_H,W ; Si. Reste el acarreo.
goto jumpIn2+2;+1

```

jumpIn2

```

Clrwdt
movf Divisor_H,W ; Parte alta del Divisor en W.

subwf DatoC_H,W ; (DatoC*10)-Divisor
movf DatoC_L,W ; DatoC es el resultado de la

```

```

; multiplicación.
movwf Acumula_L ; Pasa a Acumula_L y_H para ser
; Dividido de nuevo.
movf DatoC_H,W ; Por el divisor.
movwf Acumula_H ; La segunda división da el primer
; Decimal.
clrf Dato_32H
btfsc STATUS,C ; DatoC que es la multi. por
; 10 < Divisor?
goto $+5 ; No. Vaya a dividir el número.
clrf Cociente_L ; Si. Cociente cero.
movf Cociente_L,W ; El cociente a W.
movwf MultiC
goto $+4 ; Termina
call Division ; Llama a la división.
movf Cociente_L,W
movwf MultiC ; Resultado decimal a MultiA.
movlw .0 ; Redondeo.
iorwf MultiA,W ; Or MultiA con cero.
btfss STATUS,Z ; MultiA cero?.

```

```

goto Redondeo ; No.
movlw .0 ; Si. Or con la parte alta MultiB.
iorwf MultiB,W
btfss STATUS,Z ; MultiB cero?
goto Redondeo ; No.
goto finalex ; Si. Redondeo.

```

#### Redondeo

```

Clrwdt
movlw .5 ; Redondeo.
subwf MultiC,W ; Resto decimal de 5.
btfsc STATUS,C ; Mayor a 5?.
goto $+2 ; Si. Redondear.
goto finalex ; No. Redondear.
movlw .1
addwf MultiA ; Si. Redondeo.
btfsc STATUS,C ; Acarreo?
incf MultiB ; Si. Sume el acarreo.

```

#### finalex

```

Clrwdt
movf MultiA,W ; Carga el dato menos significativo .

```

```

call    Buffer_Salida    ; Se envía el dato digital ya a el
                                ; Maestro.

movf    MultiB,W        ; Carga el dato más significativo.

call    Buffer_Salida    ; Se envía el dato digital ya a el
                                ; Maestro.

return

```

\*\*\*\*\*

#### Despertar\_Sleep

```

bcf     PIR1,SSPIF      ; Repone flag por llegada de bit
                                ; Start.

bcf     PORTB,2

btfsc  STATUS,4

goto   $+2

btfsc  STATUS,3

goto   $+3

bsf     PORTB,1

goto   fin

```

xy

```

clrwdt

```

```

banksel SSPBUF      ; Banco0.
movf  SSPBUF,W      ; Se lee SSPBUF para que BF sea
                    ;cer0.
btfsc SSPCON,SSPOV ; Si SSPOV =1 lo limpia una línea
                    ; Después.
bcf   SSPCON,SSPOV ; Si no va a al banco 1.
Banksel SSPSTAT     ; Banco1.
btfsc SSPSTAT,BF   ; Verifica si el buffer esta lleno BF.
goto  xy
banksel ADCON1     ; Banco1.
movlw b'10000000' ; Configura A/D como entradas
                    ; Analógicas.
movwf ADCON1      ; Justificación derecha
                    ; ADRESH:ADRESL.
movlw b'11000111' ; TMR0 habilitado con predivisor 128 en
movwf OPTION_REG  ; OPTION_REG.
Clrwdt
banksel ADCON0     ; Banco0.
movlw 0x01        ; Se inicializa la adquisición del
movwf PORTB       ; Primer canal de 8 datos.

```

```

movlw b'10000001' ; Frecuencia conversión , se escoge el
; Canal

movwf ADCON0 ;ceros y se habilita el A/D con ADON=1.
; Se ingresan los datos que soporta
; El esclavo en el buffer de salida
; Nunca un número impar.

movlw .3 ; Número Datos que convierte el
movwf Number_Datos ; ADC y que Maneja El software.
Movlw .6 ; Número datos a transmitir dos x cada
; Conversión ADRESH:L.

call Buffer_Salida ; Se envía el dato al maestro.
call Promedio ; Dato De Corriente

movlw b'10011001' ; Frecuencia conversión A/D, escoge
movwf ADCON0 ; Canal tres y se habilita el A/D con
movlw .255 ; ADON =1.

call Timer

call Promedio ; DATO DE TEMPERATURA

movlw b'10100001' ; Frecuencia conversión A/D, escoge
movwf ADCON0 ; Canal cuatro y se habilita el A/D con
; ADON =1.

```

```
        call    Promedio      ; Dato De Voltaje  
fin  
        return                ; Fin.
```

\*\*\*\*\*

inicio

```
        Clrwdt  
        call    Ini_Puertos   ; Se inicializan los puertos.  
        call    Config_Puertos ; Se configuran los puertos como E/S.  
        bsf    PORTB,2  
        call    Config_I2C    ; Se configura el PIC como esclavo en  
                                ; I2C.  
        call    Interrupt_on  ; Se habilita interrupciones despertar  
                                ; Sleep.  
        movf   SSPBUF,W      ; Limpio REG. SSPBUF para que  
                                ; BF =0.  
        bcf    SSPCON,SSPOV  ; Me aseguro que SSPOV =0.  
        banksel OPTION_REG  
        movlw  b'11001111'  
        movwf  OPTION_REG  
        banksel PORTA
```

```
                clrwdt
wait
                sleep                ; Modo sleep.
                call    Despertar_Sleep ; Llegó bit START y la dirección?
                END
```

### *Puesta en Marcha*

### *Instalación del Hardware*

- 1.- Instalar en el módulo maestro cerca de la PC alimentado con una tensión de red de 120VAC.
- 2.- Conectar los cables conectores DB9. El módulo maestro tiene dos conectores DB9, un cable conector al puerto comm1 de la PC y el otro al conector DB9 del módulo Esclavo. Los módulos esclavos se conectan unos con otro con cables conectores DB9.
- 3.- Ubicar los módulos esclavos lo más cerca de cada cargador alimentado con una tensión de red de 120VAC

- 4.- Conectar las señales del monitoreo corriente, voltaje y temperatura al módulo esclavo en las borneras de conexión de señales. Cada bornera indica la señal correspondiente.
- 5.- La señal de corriente se toma del shunt del amperímetro del cargador, la señal de voltaje se toma de la salida del cargador (cable rojo positivo y cable negro negativo).
- 6.- Introducir el sensor de temperatura lm334Z en el interior de una celda de una batería en la línea, la señal del sensor se conecta en la bornera de conexión correspondiente en el módulo esclavo.

#### *Instalación Software*



- 1.- Insertar el CD de instalación en la unidad de CD y al abrir automáticamente los archivos de instalación debe iniciar el instalador Setup .
- 2.- Una vez se inicia la instalación siga los pasos a través de los cuadros de diálogos hasta finalizarla. Si no está inseguro de las opciones que indican los cuadros de diálogos, simplemente acepte las opciones por default mostradas.
- 3.- El software se instala en C:\Program Files\Sistema de Monitoreo del disco duro de la PC.

4.- En la carpeta Sistema de Monitoreo se guardan los archivos en Acces y Word para el funcionamiento del software y para la adquisición y almacenamiento de datos. Los archivos son: *Sistema\_MonitoreoAcces* y *EspecificacionBosch*.

5.- Inicie el software Sistema de Monitoreo ejecutando *Sistema\_Monitoreo.EXE* o haciendo Clic en el icono ubicado en el escritorio. Se iniciará el software con su presentación (Figura D1).

Figura D1

Icono del software Sistema de Monitoreo



### *Guía del Usuario*

1.- Iniciado el programa se observará la pantalla de Menú Principal (Figura D2), Luego hacemos clic en el Botón de configuración.

Figura D2.  
Menú Principal

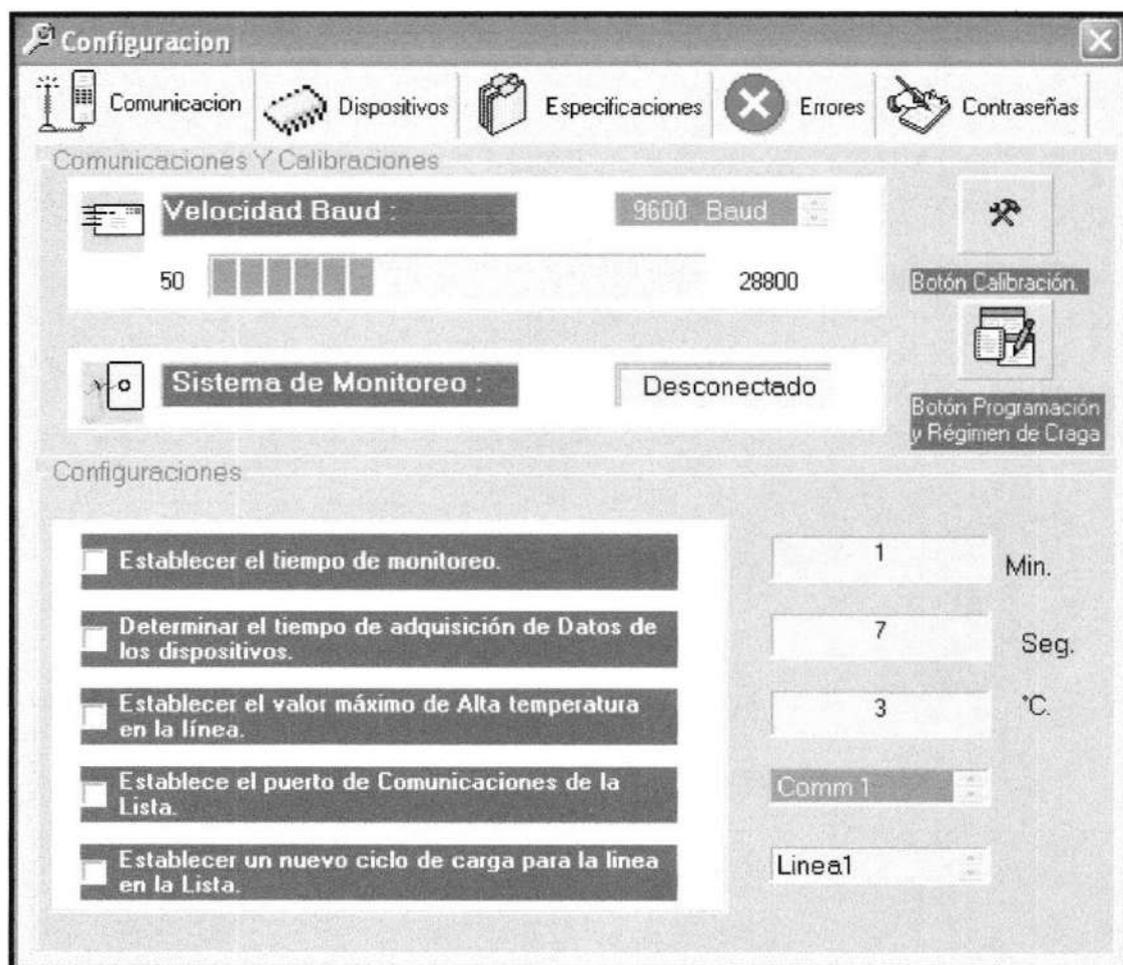


2.- En la pantalla de Configuración vamos a Comunicación > Configuraciones y establecemos en los cuadros de texto el tiempo de monitoreo, tiempo de adquisición de datos y de Alta Temperatura que queremos definir para nuestra aplicación. Luego de definidos estos tiempos hacemos clic en los cuadros de

verificación para que el sistema guarde estas configuraciones en la base de datos (Figura D3).

Figura D3

Pantalla Configuraciones.



3.- El siguiente paso es programar el Régimen de Carga de un ciclo de carga de todo el sistema. Para lo cual en la pantalla Configuraciones vamos a Comunicación > Comunicaciones y Calibraciones > Programación y Régimen de Carga escogemos de la Lista Desplegable el número de la línea de baterías, escogemos el Id del tipo de batería que se va a cargar en esa línea (Según Tabla) y lo ingresamos en el cuadro de texto. Automáticamente se observan en las etiquetas los amperios-horas y horas de carga del tipo de batería programado (Figura D4).

Figura D4

Programación y Régimen de Carga.

**Programación y Régimen de Carga.** [X]

Línea: 2

Tipo de Batería: 12/38/9

Ah: 152

Horas: 30

Nuevo ciclo de Carga para la línea en la lista.

Nuevo ciclo de carga para todas las Líneas.

Ah actuales Solo Números

Id	Tipo de Bate	Nº Placa +	Nº Placa -	Cantidad de Ah	operaje de Carga	Horas de
1	6/160/27	14	13	560	14	
2	8/195/33	17	16	0.1	17	
3	12/38/9	5	4	152	5	

5.- Una vez configurado los datos del sistema y programada las líneas de Carga, se conectan los módulos maestro-esclavos al Sistema por medio del cable conector DB9. El Sistema detecta automáticamente el número de esclavos y nos muestra un cuadro (Figura D5) donde se indica el código interno del esclavo y debemos asignar un número de línea de baterías, además datos técnicos del cargador en el que se ubica como corriente máxima y voltaje en mV del sensor de corriente (Shunt).

Figura D5

Cuadro de Detección de Esclavo.



Se ha detectado una tarjeta conectada al sistema.

Código	<input type="text"/>
Línea	<input type="text"/>
Shunt	<input type="text"/>
Max. Corriente	<input type="text"/>

OK Cancelar

El sistema una vez configurado, programado y detecta los módulos Maestro-Esclavos, puede empezar a monitorear. El usuario puede monitorear en la pantalla Monitoreo de Baterías y hacer consultas en Gráficos Estadístico, Tabla de Alarmas y Tabla de Datos.

## **E. MONTAJE Y COSTOS DEL SISTEMA DE MONITOREO**

### *Montaje*

El sistema de Monitoreo está dividido en módulos Esclavos y un módulo Maestro y una PC que analiza los datos.

El módulo Maestro se monta lo más cercano de la PC. La tarjeta Maestro consta de dos conectores DB-9 (Conexión en Cascada), que mediante un cable conector DB-9 de tres hilos conecta la tarjeta maestro con el puerto serial Comm de la PC y el otro con el cable conector DB-9 se conecta con una tarjeta Esclavo. La distancia entre la tarjeta maestro y la tarjeta esclavo es muy flexible en este proyecto se han utilizado hasta 9 metros. La distancia entre tarjetas

Esclavo máximo es de 1 metros, distancias mayores podrían crear problemas en la transmisión de datos.

Las tarjetas Esclavos también se conectan con cables conectores DB-9 unas a otras en cascada. Las tarjetas Esclavos se ubican en cada cargador de baterías. Dependiendo del número de líneas en un cargador, existirán el mismo número de tarjetas Esclavos en él.

La fuente de alimentación de las tarjetas electrónicas es de 120 voltios alternos a un transformador reductor de 120/14 voltios que va a una fuente reguladora de voltaje que mantiene el voltaje de alimentación DC en 5V, -5V, 12V. Los transformadores se montan junto a las tarjetas en el módulo y conectan su salida a las tarjetas por medio de borneras de conexiones soldadas en ellas.

Las señales que se monitorean se toman de los cargadores en el caso de la corriente y el voltaje, y en el caso de la temperatura en las baterías. La señal de corriente es tomada de los terminales de los amperímetros de aguja de los

cargadores, son tomadas con cable #16 AWG y terminales de ojo para conectarse a los terminales del amperímetro. La señal de Voltaje es tomada de la salida de voltaje del cargador de baterías (Cable rojo-negro).

El amperímetro se encuentra en el cable positivo del cargador de baterías, el terminal izquierdo es el positivo y es la señal de corriente mientras que el terminal derecho es la referencia analógica de nuestro sistema. La señal de voltaje se toma del cable negativo de la salida del cargador, cuya señal es tomada con cable #16 AWG a una bornera de conexión de la tarjeta Esclavo. La temperatura es tomada del sensor de temperatura introducido en cualquiera de las celdas de la línea de baterías. El sensor se encuentra en un capuchón de teflón y con cable apantallado transmite la señal de corriente proporcional a la temperatura algunos metros desde el cuarto de carga al cuarto de cargadores donde se encuentra la tarjeta Esclavo.

#### *Costos del Sistema de Monitoreo*

Los Costos del sistema se dividen en Módulo Esclavo y Maestro, el costo fijo es del módulo Maestro, así los costos del sistema de monitoreo dependen del

número de módulos Esclavo a instalar. En este proyecto se instalaron dos módulos esclavos.

COSTOS DEL SISTEMA DE MONITOREO			
<b>Módulo:</b>	Esclavo		
<b>Ubicación:</b>	Cargadores		

Elementos	Cantidad	P.Unitario	Total
R 56k,1/2W	1	0,03	0,03
R 6,8 M,1/2W	1	0,05	0,05
R 10k,1/2W	3	0,03	0,09
R 1k,1/2W	3	0,03	0,09
R 220,1/2W	1	0,03	0,03
R 330,1/2W	4	0,03	0,12
R 4,7k,1/2W	1	0,03	0,03
Capac.0,1uf,Cerámico	4	0,06	0,24
Capac.100uf,25V	1	0,05	0,05
Capac.22 pF,Cerámico	2	0,12	0,24
Capac.470uf,25V	1	0,12	0,12
zocalos 6p	1	0,1	0,1
zocalos 14p	1	0,15	0,15
zocalos 40p ancho	1	0,18	0,18
Diodo 4N007	1	0,1	0,1
Relay 12 Vdc	1	1,5	1,5
Puente Rec 1A	2	0,3	0,6
LM741	1	0,35	0,35
Optocupla 4N35	1	0,43	0,43
Lm7805	1	0,4	0,4
Lm7812	1	0,4	0,4
Lm7905	1	0,4	0,4
Borneras 3 Esp.	3	0,85	2,55
Borneras 2 Esp.	1	0,85	0,85
Botonera	1	0,1	0,1
Cristal 20MHz	1	0,65	0,65
Diseño Tarjeta Electrónica	1	15	15
Trimer 10k	1	1,9	1,9
Conector DB-9 hembra	2	0,35	0,7
Conector DB-9 macho	2	0,35	0,7
LM336	1	1	1
PIC16F877A	1	9	9
Transformador 120/14 VAC	2	6	12

Transistor 2N3904	1	0,4	0,4
Tornillos-tuercas	1	3	3
		Subtotal	67,1
		Iva%	8,052
		<b>TOTAL</b>	<b>75,152</b>

COSTOS DEL SISTEMA DE MONITOREO			
<b>Módulo:</b>	Maestro		
<b>Ubicación:</b>	PC		

Elementos	Cantidad	P.Unitario	Total
R 1,5k, 1/2W	3	0,03	0,09
Fusible 1[A]	1	0,4	0,4
Capac.0,1uf,Cerámico	1	0,06	0,06
Capac.22 pF,Cerámico	2	0,12	0,24
Capac.470uf,25V	1	0,12	0,12
zocalos 16p	1	0,15	0,15
zocalos 40p ancho	1	0,18	0,18
Puente Rec 1A	1	0,3	0,3
Lm7805	1	0,4	0,4
Borneras 2 Esp.	1	0,85	0,85
Botonera	1	0,1	0,1
Cristal 20MHz	1	0,65	0,65
Diseño Tarjeta Electrónica	1	8	8
Conector DB-9 hembra	2	0,35	0,7
Conector DB-9 macho	2	0,35	0,7
Maxim Rs-232	1	1	1
PIC16F877A	1	9	9
Transformador 120/14			
VAC	1	6	6
PC Pentium 3	1	500	500
		Subtotal	528,94
		Iva%	63,47
		<b>TOTAL</b>	<b>592,41</b>

## BIBLIOGRAFIA.

### LIBROS

- 1 JOSÉ MARIA ANGULO U., IGNACIO ANGULO M, Microcontroladores Pic. Diseño práctico de aplicaciones (1.ª parte: PIC16F84), Editorial McGraw-Hill 1ª edición 1997.
- 2 JOSÉ MARIA ANGULO, SUSANA ROMERO, IGNACIO ANGULO M. Microcontroladores Pic. PIC16F877A. Diseño práctico de aplicaciones (2.ª parte: PIC16F877). Editorial Mc Graw-Hill 2ª edición 2000.
- 3 LITTLE GIANT. C - Programmable Miniature Controller. Technical Reference Manual. Z-World. Inc. Edición Z-world. Inc ,1ª edición 1990-1993.
- 4 Visual Basic 6.0. Manual de Referencia. Editorial McGraw-Hill.
- 5 NELS E. HEHNER. Manual de Fabricación de Acumuladores. Editorial Arpa Editores. 3ª edición 1985.

## INTERNET

Microchip Technology Inc.	<a href="http://www.microchip.com">http://www.microchip.com</a>
Microsystems Engineering	<a href="http://www.arrakis.es/~msyseng">http://www.arrakis.es/~msyseng</a>
Parallax, Inc.	<a href="http://www.parrallaxinc.com">http://www.parrallaxinc.com</a>
Itu Technologies	<a href="http://www.itutech.com">http://www.itutech.com</a>
Sagitron	<a href="http://www.sagitron.es">http://www.sagitron.es</a>
Orcad	<a href="http://www.orcad.com/technical/technical.asp">http://www.orcad.com/technical/technical.asp</a>