

AUTOMATIZACIÓN DEL SISTEMA DE CONTROL DE TIEMPO DEL  
TRANSPORTE DE BUSES URBANOS

ROMMEL OSWALDO CABRERA VELASCO  
JULIO CÉSAR LASSO RODRÍGUEZ

UNIVERSIDAD DE NARIÑO  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA ELECTRÓNICA  
SAN JUAN DE PASTO  
2008

AUTOMATIZACIÓN DEL SISTEMA DE CONTROL DE TIEMPO DEL  
TRANSPORTE DE BUSES URBANOS

ROMMEL OSWALDO CABRERA VELASCO  
JULIO CÉSAR LASSO RODRÍGUEZ

Trabajo de grado presentado como requisito parcial para optar el título de  
Ingeniero Electrónico

Director de Proyecto:  
JOSÉ DOLORES RODRÍGUEZ MARTÍNEZ  
Ingeniero Electrónico

UNIVERSIDAD DE NARIÑO  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA ELECTRÓNICA  
SAN JUAN DE PASTO  
2008

“Las ideas y conclusiones aportadas en la tesis de grado, son responsabilidad exclusiva de los autores”

Artículo 1º. Acuerdo No. 324 de octubre 11 de 1966, emanado del Honorable Consejo Directivo de la Universidad de Nariño.

Nota de aceptación:

---

---

---

---

---

---

---

---

Firma del Presidente del Jurado

---

Firma del Jurado

---

Firma del Jurado

San Juan de Pasto, Abril de 2008

A Dios Todopoderoso, por regalarme la vida, por ser mí guía, la luz que ilumina mi camino y por todas las oportunidades que me brinda.

A mis Padres, por su dedicación, paciencia, amor, por su apoyo, su confianza depositada en mi, por pensar en mi bienestar antes que en el de ellos, por darme el mejor regalo del mundo “la educación”; a ustedes les debo este logro, mil gracias.

A mis hermanos, a mi abuelita y a mis familiares, que han estado siempre conmigo apoyándome a alcanzar mis metas.

A mi segundo hogar la familia Enríquez Martínez, por acogerme como un integrante más, por su apoyo, por aguantarme durante todo este tiempo y por sus sabios consejos que espero seguir el resto de mi vida.

A mi compañero Julio Cesar por ser excelente persona y amigo.

Al mejor país del mundo.... **“Mi Colombia”**

**Rommel Cabrera V**

A Dios por estar a mi lado siempre, Él me ha brindado lo mejor de la vida...

A mis padres: Victoria y Felix quienes me han amado  
y me han guiado por el camino correcto.

A mi esposa Andrea, mi compañera incondicional, quien siempre  
me ha brindado su inmenso amor y su gran apoyo.

A mi hija Angie Vanessa a quien adoro con toda mi alma.

A Rommel Cabrera, compañero de trabajo y amigo fiel.

A todos mis familiares y amigos.

**Julio César Lasso R.**

## **AGRADECIMIENTOS**

Los Autores expresan sus agradecimientos a:

Ing. José Dolores Rodríguez, Director del trabajo de grado por su constante dedicación, orientación y su interés en llevar a cabo el proyecto.

A la Empresa Transportadora Autobuses del Sur, en especial a su gerente Dr. Guillermo Valencia, por su valiosa colaboración.

Ing. Vicente Aux, por su tiempo, apoyo, dedicación, paciencia, por ayudarnos cuando más lo necesitábamos.

Ing. Darío Fajardo, por su atención y colaboración, por resolver nuestras dudas en cualquier momento.

Al estudiante Luís Felipe Villa, por sus consejos que fueron de gran ayuda para el desarrollo del proyecto.

A Oscar Delgado, por su amistad y colaboración.

A todos los profesores del departamento de Electrónica, por sus conocimientos brindados.

Y a todos nuestros compañeros y amigos que contribuyeron de alguna manera en la realización del presente trabajo.

## CONTENIDO

	Pág.
INTRODUCCIÓN	24
1. MARCO CONCEPTUAL	26
1.1 SISTEMAS DE COMUNICACIÓN	26
1.1.1 Elementos de un sistema de comunicaciones	26
1.1.2 Comunicaciones de datos	27
1.1.3 Transmisión de datos analógicos y digitales	28
1.1.4 Perturbaciones en la transmisión	29
1.1.5 Ancho de banda (BW)	30
1.1.6 Medios de transmisión	31
1.1.7 Espectro electromagnético	32
1.1.8 Modulación	33
1.1.9 Técnicas de modulación digital con portadora modulada	34
1.2 INTERFACES DE COMUNICACIÓN SERIAL	35
1.2.1 Norma RS232	36
1.2.2 El Bus Serie Universal (USB)	37
1.3 MICROCONTROLADORES	42
2. DESARROLLO	44
2.1 DESCRIPCIÓN GENERAL	44
2.1.1 Módulo fijo	44
2.1.2 Módulo móvil	45

	Pág.
2.1.3 Módulo de configuración	46
2.2 DESCRIPCIÓN DEL HARDWARE	46
2.2.1 Selección de microcontroladores	46
2.2.1.1 Microcontrolador dispositivo fijo y unidad de alimentación	47
2.2.1.2 Microcontrolador dispositivos móvil y de configuración	48
2.2.2 Selección módulos de radio frecuencia	50
2.2.2.1 Módulo de transmisión	50
2.2.2.2 Módulo de recepción	52
2.2.3 RTC (Reloj de Tiempo Real)	53
2.2.4 Memoria externa serial EEPROM	57
2.3 DESARROLLO DE SOFTWARE Y CONFIGURACION INTERNA DE LOS MICROCONTROLADORES	60
2.3.1 Programación y configuración módulo fijo	60
2.3.1.1 Desarrollo de software implementado en microcontrolador	60
2.3.1.2 Configuración de registros del microcontrolador	64
2.3.2 Programación y configuración módulo de configuración	66
2.3.2.1 Desarrollo de software implementado en microcontrolador	66
2.3.2.2 Configuración de registros del microcontrolador	74
2.3.3 Programación y configuración módulo móvil	77
2.3.3.1 Desarrollo de software implementado en microcontrolador	78
2.3.3.2 Configuración de registros del microcontrolador	83
2.3.4 Programación y configuración microcontrolador del sistema de alimentación	87

	Pág.
2.3.4.1 Desarrollo de software implementado en microcontrolador	88
2.3.4.2 Configuración de registros del microcontrolador	91
2.4. FIRMWARE, DRIVER, SOFTWARE Y PROCEDIMIENTOS DE INSTALACIÓN	93
2.4.1 Instalación del dispositivo móvil	96
2.4.2 Desarrollo de la aplicación	99
3. PRUEBAS E IMPLEMENTACIÓN	103
3.1 PRUEBAS PRELIMINARES	108
3.2 CORRECCION DE ERRORES Y PRUEBAS FINALES	109
4. CONCLUSIONES	112
5. RECOMENDACIONES	113
BIBLIOGRAFIA	114
ANEXOS	116

## LISTA DE TABLAS

	Pág.
Tabla 1. Clases de modulación	33
Tabla 2. Especificaciones microcontrolador PIC16F628A	47
Tabla 3. Especificaciones microcontrolador PIC18F2550	49
Tabla 4. Características principales del transmisor TLP434A	50
Tabla 5. Bandas de frecuencias para aplicaciones de telemetría y telecontrol con bajos niveles de potencia	51
Tabla 6. Características principales del receptor RLP434A	52
Tabla 7. Características principales Antena WLP-434 RF	52
Tabla 8. Características principales del RTC DS1307	53
Tabla 9. Registros del reloj de tiempo real DS1307	55
Tabla 10. Configuración del registro de Control	56
Tabla 11. Características de memoria EEPROM 24LS256.	58
Tabla 12. Configuración del registro CONFIG	64
Tabla 13. Rango de valores de frecuencia y capacitancia	65
Tabla 14. Configuración del registro TXSTA	65
Tabla 15. Configuración del registro RCSTA	65
Tabla 16. Configuración del registro SPBRG	65
Tabla 17. Configuración del registro CONFIG1H	75
Tabla 18. Configuración del registro CONFIG1L	75
Tabla 19. Configuración del registro TXSTA	76

	Pág.
Tabla 20. Configuración del registro RCSTA	76
Tabla 21. Configuración del registro BAUDCON	76
Tabla 22. Configuración del registro SPBRG	77
Tabla 23. Configuración del registro CONFIG1L	84
Tabla 24. Configuración del registro CONFIG1H	84
Tabla 25. Configuración del registro OSCCON	85
Tabla 26. Configuración del registro UCON	85
Tabla 27. Configuración del registro UCFG	87
Tabla 28. Configuración del registro CONFIG	92
Tabla 29. Configuración del registro PCON	92
Tabla 30. Configuración del registro CMCOM	92

## LISTA DE FIGURAS

	Pág.
Figura 1. Elementos de un sistema de comunicación	27
Figura 2. Modelo simplificado para las comunicaciones de datos	27
Figura 3. Efecto del ruido en una señal digital	30
Figura 4. Comunicación direccional	31
Figura 5. Comunicación omnidireccional	32
Figura 6. Espectro electromagnético para las telecomunicaciones	32
Figura 7. Modulación	33
Figura 8. Modulación ASK	34
Figura 9. Modulación FSK	34
Figura 10. Modulación PSK	35
Figura 11. Estructura de un caracter que se trasmite en forma serial Asíncrono	35
Figura 12. Estructura de capas del bus usb	37
Figura 13. Flujo de comunicaciones entre dispositivo usb y el host	38
Figura 14. Codificación NRZI	40
Figura 15. Señal sync de sincronismo para baja velocidad	40
Figura 16. Esquema del cable USB	41
Figura 17. Diagrama de bloques del módulo fijo	44
Figura 18. Diagrama de bloques del módulo móvil	45
Figura 19. Diagrama de bloques del módulo de configuración	46

	Pág.
Figura 20. Encapsulado PIC16F628A	47
Figura 21. Encapsulado PIC18F2550	49
Figura 22. Grafico corriente vs voltaje y potencia vs voltaje	51
Figura 23. Distribución de pines del RTC DS1307	53
Figura 24. Dirección del RTC como esclavo en el bus I <sup>2</sup> C	54
Figura 25. Diagrama de conexión maestro (microcontrolador), Esclavo (RTC)	54
Figura 26. Protocolo de escritura del maestro (microcontrolador) Sobre el esclavo (RTC)	55
Figura 27. Protocolo de lectura del maestro (microcontrolador) Sobre el esclavo (RTC)	56
Figura 28. Encapsulado memoria EEPROM 24LS256	57
Figura 29. Formato del byte de control memoria EEPROM 24LS256	58
Figura 30. Lectura aleatoria de datos	58
Figura 31. Lectura secuencial de datos	59
Figura 32. Escritura de un byte	59
Figura 33. Escritura de varios bytes	59
Figura 34. Diagrama eléctrico módulo fijo	60
Figura 35. Diagrama de flujo rutina principal microcontrolador módulo fijo	61
Figura 36. Diagrama de flujo subrutina transmisión de datos	62
Figura 37. Mapa de memoria EEPROM PIC16F628A	62
Figura 38. Diagrama de flujo subrutina configuración RTC	63
Figura 39. Diagrama eléctrico módulo de configuración	66

	Pág.
Figura 40. Diagrama de flujo rutina principal microcontrolador módulo de configuración	67
Figura 41. Mapa de memoria EEPROM PIC18F2550	68
Figura 42. Diagrama de flujo subrutina configurar RTC (a)	69
Figura 43. Diagrama de flujo subrutina configurar RTC (b)	70
Figura 44. Diagrama de flujo subrutina cambio de código de área	71
Figura 45. Diagrama de flujo subrutina actualizar módulo fijo	72
Figura 46. Diagrama de flujo subrutina leer datos módulo fijo	73
Figura 47. Diagrama de flujo subrutina estado de configuración	74
Figura 48. Diagrama eléctrico módulo móvil	77
Figura 49. Diagrama de flujo rutina principal microcontrolador módulo móvil	78
Figura 50. Diagrama de flujo de la subrutina recibir	79
Figura 51. Parte inicial del mapa de memoria de externa 24LS256	80
Figura 52. Diagrama de flujo de la subrutina grabar	81
Figura 53. Diagrama de flujo de la interrupción usb_conect	82
Figura 54. Datos recibidos por el PC	83
Figura 55. Periférico USB y opciones	86
Figura 56. Diagrama eléctrico UPS	87
Figura 57. Diagrama de flujo rutina principal microcontrolador de UPS	88
Figura 58. Mapa de memoria EEPROM PIC16F628A	89
Figura 59. Diagrama de flujo subrutina contador de descarga	89
Figura 60. Diagrama de flujo subrutina cargar batería	90

	Pág.
Figura 61. Conexión interna de módulo de comparadores	93
Figura 62. Instalación de nuevo hardware (a)	96
Figura 63. Instalación de nuevo hardware (b)	96
Figura 64. Instalación de nuevo hardware (c)	97
Figura 65. Instalación de nuevo hardware (d)	97
Figura 66. Instalación de nuevo hardware (e)	98
Figura 67. Instalación de nuevo hardware (f)	98
Figura 68. Seguridad de acceso	99
Figura 69. Aplicación principal para el control de buses	99
Figura 70. Ventana control de datos	100
Figura 71. Visualización de datos	100
Figura 72. Guardar datos en el pc	101
Figura 73. Visualización datos en Microsoft Excel	101
Figura 74. Vista interior módulo fijo	103
Figura 75. Vista interior módulo móvil	104
Figura 76. Vista exterior módulo móvil	104
Figura 77. Vista interior módulo de configuración	105
Figura 78. Vista exterior módulo de configuración	105
Figura 79. Barrió El Rosario (diagonal 16, cra 13 este), código 10	106
Figura 80. Barrio Miraflores (calle 16D, cra 2E), código 20	106
Figura 81. Avenida las Américas (Av Américas, calle 13), código 30	107
Figura 82. Universidad de Nariño (calle 18, cra 50), código 40	107

	Pág.
Figura 83. Iglesia San Sebastián (calle 21, cra 21), código 50	108
Figura 84. Datos recolectados en primeras pruebas	109
Figura 85. Datos recolectados en pruebas finales	110
Figura 86. Tarjeta de control de tiempos bus 75	111
Figura 87. Datos recolectados bus 75	111

## LISTAS DE ANEXOS

	Pág.
ANEXO A. Especificaciones técnicas de dispositivos: módulo fijo, módulo de configuración y módulo móvil.	117
ANEXO B. Placa de circuito impreso módulo fijo. Vista componentes 3D, cara de componentes y cara de pistas	119
ANEXO C. Placa de circuito impreso módulo de configuración: Vista componentes 3D, cara de componentes y cara de pistas	120
ANEXO D. Placa de circuito impreso módulo móvil: Vista componentes 3D, cara de componentes y cara de pistas	121
ANEXO E. Placa de circuito impreso pantalla LCD: Vista componentes 3D, cara de componentes y cara de pistas	122
ANEXO F. Placa de circuito impreso UPS módulo fijo: Vista componentes 3D, cara de componentes y cara de pistas	123
ANEXO G. Placa de circuito impreso UPS módulo móvil: Vista componentes 3D, cara de componentes y cara de pistas	124
ANEXO H. Diagrama del módulo oscilador dispositivo de configuración	125
ANEXO I. Diagrama del módulo oscilador dispositivo móvil	126
ANEXO J. Mapa de la ciudad San Juan de Pasto, en el que se identifican la ubicación de los puntos de control y el recorrido de la ruta N° 12 de Autobuses	127
ANEXO K. Código fuente microcontrolador módulo fijo (Archivo Anexo_K.pdf en CD)	
ANEXO L. Código fuente microcontrolador módulo de configuración (Archivo Anexo_L.pdf en CD)	
ANEXO M. Código fuente microcontrolador módulo móvil (Archivo Anexo_M.pdf en CD)	

ANEXO N. Código fuente librería USB\_CDC\_ConBuses (Archivo Anexo\_N.pdf en CD)

ANEXO O. Código fuente microcontrolador UPS (Archivo Anexo\_O.pdf en CD)

ANEXO P. Código fuente aplicación manejo de datos PC (Archivo Anexo\_P.pdf en CD)

ANEXO Q. Resolución número 000797 de 8 de junio de 2001, Ministerio de Comunicaciones Colombia (Archivo Anexo\_Q.pdf en CD)

ANEXO R. Datasheet microcontrolador PIC18F2550 (Archivo Anexo\_R.pdf en CD)

ANEXO S. Datasheet microcontrolador PIC16F628A (Archivo Anexo\_S.pdf en CD)

ANEXO T. Datasheet RTC DS1307 (Archivo Anexo\_T.pdf en CD)

ANEXO U. Datasheet memoria 24LC256 (Archivo Anexo\_U.pdf en CD)

ANEXO V. Datasheet transmisor TLP434A (Archivo Anexo\_V.pdf en CD)

ANEXO W. Datasheet receptor RLP434 (Archivo Anexo\_W.pdf en CD)

ANEXO X. Datasheet antena WLP434 (Archivo Anexo\_X.pdf en CD)

## GLOSARIO

**BCD:** (Binary Coded Decimal), sistema numérico usado en sistemas computacionales y electrónicos para codificar números enteros positivos y facilitar las operaciones aritméticas. Es un código pesado debido a que representa los dígitos con un orden específico (8421).

**BPS:** bits por segundo, en una transmisión de datos, es el número de impulsos elementales (1 ó 0) transmitidos en cada segundo. Los bits por segundo como unidad del SI (sistema internacional) son utilizados para expresar la velocidad de transmisión de datos o bit rate.

**COMUNICACIÓN SERIAL:** proceso de envío de datos de un bit, secuencialmente, sobre un canal de comunicación o un bus de computadora.

**DRIVER:** programa que interactúa con un dispositivo en particular. Contiene todos los datos necesarios del dispositivo con el que se entrega para que el resto de programas sepan como han de utilizarlo. Normalmente, los drivers son DLL's.

**EEPROM:** son las siglas de electrically-erasable programmable read-only memory (ROM programable y borrable eléctricamente). Es un tipo de memoria ROM que puede ser programado, borrado y reprogramado eléctricamente, a diferencia de la EPROM que solo se puede borrar mediante rayos ultravioletas. Aunque una EEPROM puede ser leída un número ilimitado, sólo puede ser borrada y reprogramada entre 100.000 y 1.000.000 veces.

**ESCLAVO:** dispositivo secundario.

**FIRMWARE:** bloque de instrucciones de un programa para propósitos específicos, grabado en una memoria tipo ROM, que establece la lógica de más bajo nivel que controla los circuitos electrónicos de un dispositivo de cualquier tipo. Al estar integrado en la electrónica del dispositivo es en parte hardware, pero también es software, ya que proporciona lógica y se dispone en algún tipo de lenguaje de programación. Funcionalmente, el firmware es el intermediario (interfaz) entre las órdenes externas que recibe el dispositivo y su electrónica, ya que es el encargado de controlar a ésta última para ejecutar correctamente dichas órdenes externas

**HOST:** equipo anfitrión.

**HUB:** dispositivo que permite centralizar el cableado de una red y poder ampliarla. Este recibe una señal y la repite emitiéndola por sus diferentes puertos.

I2C: bus de comunicaciones serie. Su nombre viene de Inter-Integrated Circuit (Circuitos Inter-Integrados).

LCD: (Liquid Crystal Display), pantalla de cristal líquido, delgada y plana formada por un número de píxeles en color o monocromos colocados delante de una fuente de luz o reflectora.

MAESTRO: dispositivo primario (equivalente a host).

NRZI: (Non Return to Zero Inverted), variante del código NRZ, se representa únicamente los "1" por cambio en la polaridad del medio magnético, que se traduce en un impulso durante la operación de lectura. La inexistencia de señal se interpreta como un "0".

PC: (personal computer) computador personal.

PROTOCOLO DE COMUNICACIÓN: conjunto de reglas que especifican el intercambio de datos u órdenes durante la comunicación entre las entidades que forman parte de una red.

RESET: puesta en condiciones iniciales de un sistema. Este puede ser mecánico, electrónico o de otro tipo.

RF: radio frecuencia.

RTC: (real time clock), circuito integrado reloj – calendario.

SENSAR: anglicismo equivalente al verbo detectar.

TRANSCEIVER: en comunicaciones (informática) es un transmisor/receptor de señales de radio frecuencia (RF), sirve para conectar aparatos por vía inalámbrica.

UPS: (Uninterruptible Power Supply), dispositivo que, gracias a su batería, puede proporcionar energía eléctrica tras un apagón a todos los aparatos conectados a la red eléctrica.

USART: (Universal Synchronous/Asynchronous Receiver Transmitter), periférico para la transmisión de datos en formato serie, usa técnicas de transmisión sincrónica o asincrónica, según se configure el periférico.

USB: (Universal Serial Bus), puerto que sirve para conectar periféricos a una computadora.

## RESUMEN

Para mejorar el sistema actual de control manual de tiempo de los buses urbanos se diseña un conjunto de dispositivos electrónicos basados en microcontroladores para realizar dicha tarea de manera automática. Para la transmisión y recepción de los datos se utilizan los módulos de radio frecuencia TLP434A y RLP434 con modulación ASK y sintonizados a 433.92MHz, el protocolo de comunicación utilizado es el RS-232.

El sistema automático de control de tiempo desarrollado consta de tres módulos:

*Módulo Fijo.* Estos dispositivos se ubican en los puntos de control del recorrido de la ruta de buses. Son los encargados de enviar datos de hora, fecha, y lugar donde se encuentra cada uno de ellos, esta función se realiza vía RF haciendo uso de la banda civil de 433.9Mhz y una potencia máxima de 10mW.

*Módulo Móvil.* Este dispositivo se encuentra anclado al vehículo. Su función es la de recibir la información que envía el módulo fijo, la procesa y almacena en su memoria EEPROM. Para la visualización de los datos recolectados se capturan desde un computador mediante interfaz USB.

*Módulo de Configuración.* Este dispositivo es portátil. Su tarea es sincronizar el reloj calendario (RTC) del módulo fijo y además asignarle el lugar de control, para ello se cuenta con una interfaz RS232 en los dispositivos.

*Palabras claves:* interfaz USB, memoria EEPROM, microcontrolador, modulación ASK, protocolo RS-232, radio frecuencia, real time clock (RTC).

## ABSTRACT

In order to improve the present system of manual control of time of the urban buses a set of electronics based on microcontroller was designed to make this task of automatic way. For the transmission and reception of the data the radio frequency modules are used TLP434A and RLP434 with tuned modulation ASK and to 433.92MHz, the RS-232 communication protocol is used.

The automatic system of control developed consists of three modules:

*Fixed Module.* These devices are located them in the control posts of the route of the bus route. They are the ones in charge to send hour data, date, and place where is each one of them, this function is made via RF doing use of the civil band of 433.9Mhz and a maximum power of 10mW.

*Móvil Module.* This device is anchored to the vehicle. Its function is to receive the information that it sends the fixed module, it processes and stores in its memory EEPROM. For the visualization of the collected data they are captured from a computer by means of interface USB.

*Módulo de Configuración.* This device is portable. Its task is to synchronize the calendar clock (RTC) of the fixed module and in addition to assign the control place to him, for it is counted with an interface RS232 in the devices.

*Key words:* ASK modulation, EEPROM memory, microcontroller, radio frequency, real time clock (RTC). RS-232 protocol, USB interface.

## INTRODUCCIÓN

El crecimiento gradual de las ciudades trae consigo nuevos problemas, uno de ellos es la calidad del servicio de transporte público. Para solucionar este problema las cooperativas de transporte en la región utilizan el control manual sobre cada ruta, mediante la ubicación de personas en diferentes tramos del recorrido el cual toma y verifica el tiempo del vehículo al pasar por el control.

El método que se está usando actualmente presenta deficiencias debido al número reducido de estaciones de control. Para mejorar este sistema es necesario aumentar su número, lo que incrementa significativamente el gasto para las empresas de transporte.

Por otra parte, estos sistemas son una de las principales fuentes de caos en el tránsito de la ciudad por los excesos de velocidad que presentan los automotores antes de llegar al punto de control y la baja velocidad e inclusive el parqueo después de pasar la verificación en dicho lugar, esto también genera mala atención y a la vez peligro a los usuarios, además, existe un riesgo de accidentalidad para el personal que realiza la verificación del tiempo, ya que en muchas ocasiones cruzan imprudentemente las calles con el fin de hacer más rápido el proceso.

Esta investigación pretende mejorar el sistema de control y para ello es necesario diseñar e implementar dispositivos que permitan controlar el tiempo que debe emplear un vehículo de transporte urbano en cada tramo de la ruta establecida de manera automática, aprovechando el gran desarrollo en el diseño de sistemas de comunicación basados en la transmisión digital de datos utilizando los diferentes medios de propagación. Dada la tendencia mundial del acceso inmediato a la información en cualquier momento y lugar, las comunicaciones inalámbricas son una de las herramientas con mayor auge en la actualidad. Haciendo uso de las ventajas que brindan los avances tecnológicos en el área de comunicaciones inalámbricas, se han desarrollado aplicaciones que permitan acceder a la información de manera oportuna y confiable.

En las ciudades principales del país, como Bogotá algunas empresas de transporte urbano, intermunicipal y de carga poseen un sistema de monitoreo haciendo uso de tecnología GPS (Sistema de Posicionamiento Global), que permite la localización de cada vehículo en tiempo real. La implementación de un sistema GPS es viable por su exactitud y rentabilidad cuando se desea cubrir una extensa área con un gran número de usuarios.

La realización de este proyecto permite hacer posteriores investigaciones para su mejoramiento tanto en el área de electrónica como en sistemas informáticos.

En este documento se plantean los conceptos teóricos necesarios para la elaboración del sistema propuesto. De igual forma, se presenta el desarrollo para la elaboración del proyecto, indicando cada una de las etapas de acuerdo a la tarea que cada una desempeña. Finalmente, se presentan los resultados y las conclusiones obtenidas a partir de las pruebas y su análisis.

## 1. MARCO CONCEPTUAL

### 1.1 SISTEMAS DE COMUNICACIÓN.

Un sistema de comunicación es la combinación de circuitos y dispositivos necesarios para transmitir y recibir información de un lugar a otro; es la totalidad de mecanismos que proporcionan el enlace para la información entre fuente y destino. El objetivo es entonces proporcionar una réplica aceptable del mensaje en su destino, proporcionando para ello reducir tiempo y esfuerzo, capturar datos en su propia fuente, centralizar el control, aumentar la velocidad de entrega de la información, reducir costos de operación y de captura de datos, aumentar la capacidad de las organizaciones, a un costo incremental razonable, aumentar la calidad y la cantidad de la información y mejorar el sistema administrativo.

**Comunicación.** Es el proceso por medio del cual la información se transfiere de un punto llamado fuente, en el espacio y tiempo, a otro punto que es el destino.

**Información o Mensaje.** Es la manifestación física producida por la fuente. Se puede establecer que el mensaje producido por una fuente no es siempre eléctrico y, por lo tanto, es necesario un transductor de entrada, que convierte el mensaje en una señal; una magnitud eléctrica variable, tal como un voltaje o una corriente, y simultáneamente otro transductor en el destino, que convierte la señal de salida a la forma apropiada del mensaje.

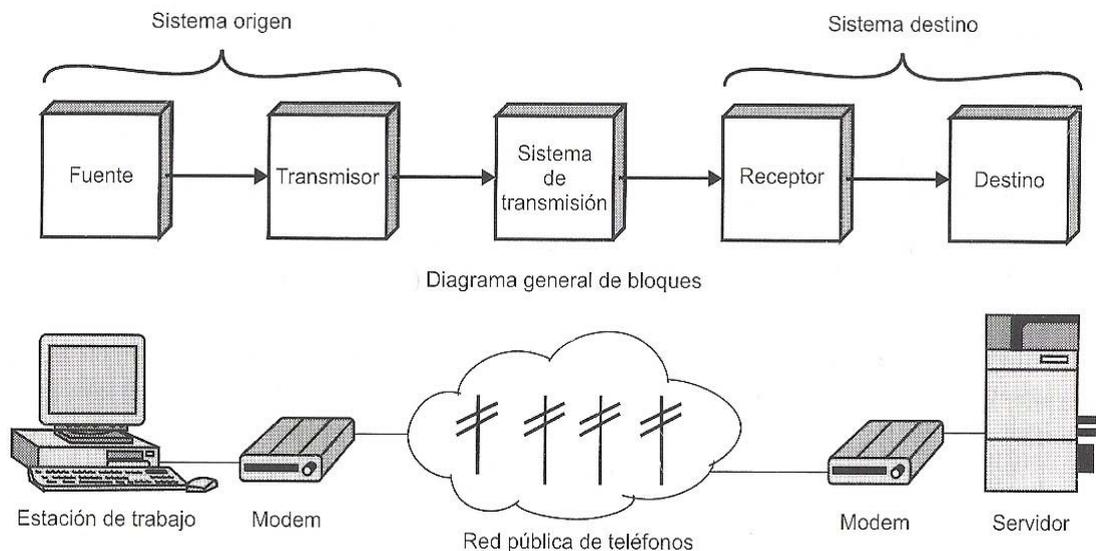
Se entiende por transmisión de datos el movimiento de información codificada, de un punto a uno o más puntos, mediante señales eléctricas, ópticas, electro-ópticas o electromagnéticas.

**1.1.1 Elementos de un sistema de comunicaciones.**<sup>1</sup> El objetivo principal de todo sistema de comunicaciones es intercambiar información entre dos entidades, la figura 1, muestra un ejemplo particular de comunicación entre una estación de trabajo y un servidor a través de una red telefónica pública. Otro ejemplo consiste en el intercambio de señales de voz entre dos teléfonos a través de la misma red anterior. Los elementos clave en este modelo son: fuente, transmisor, canal o sistema de transmisión, receptor y destino.

---

<sup>1</sup>STALLINGS, William. Comunicaciones Y Redes De Computadores. 6 ed. España: Prentice Hall, 2000, p. 4.

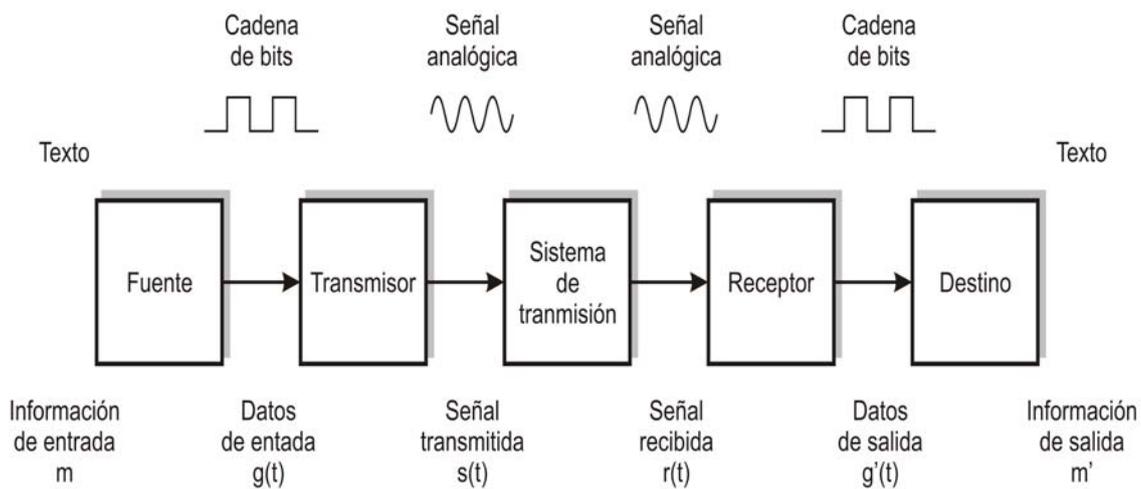
Figura 1. Elementos de un sistema de comunicación



Fuente: STALLINGS, William. Comunicaciones y Redes De Computadores. 6 ed. España: Prentice Hall, 2000.

### 1.1.2 Comunicaciones de datos.<sup>2</sup>

Figura 2. Modelo simplificado para las comunicaciones de datos



Fuente: STALLINGS, William. Comunicaciones y Redes De Computadores. 6 ed. España: Prentice Hall, 2000.

<sup>2</sup>Ibid., p. 7.

Suponiendo que tanto el dispositivo de entrada como el transmisor están en un computador personal y el usuario de dicho PC desea enviar el mensaje  $m$  a otro. El usuario activa la aplicación de correo en el PC y compone el mensaje con el teclado (dispositivo de entrada). La cadena de caracteres se almacenará temporalmente en la memoria principal como una secuencia de bits ( $g$ ). El computador se conecta a algún medio de transmisión, (una red local o una línea telefónica), a través de un dispositivo de E/S (transmisor), «transceiver» a una red local o modem. Los datos de entrada se transfieren al transmisor como una secuencia de niveles de tensión  $[g(t)]$  que representan los bits en algún tipo de bus de comunicaciones o cable. El transmisor se conecta directamente al medio y convierte la cadena  $[g(t)]$  en la señal a transmitir  $[s(t)]$ .

Al transmitir  $s(t)$  a través del medio, antes de llegar al receptor, aparecerán una serie de dificultades. Por lo tanto, la señal recibida  $r(t)$  puede diferir de alguna manera de la transmitida  $s(t)$ . El receptor intentará estimar la señal original  $s(t)$ , a partir de la señal  $r(t)$  y de su conocimiento acerca del medio, obteniendo una secuencia de bits  $g'(t)$ . Estos bits se envían al computador de salida, donde se almacenan temporalmente en memoria como un bloque de bits ( $g'$ ). En muchos casos, el destino intentará determinar si ha ocurrido un error, y en su caso, cooperar con el origen para eventualmente conseguir el bloque de datos completo y sin errores. Los datos, finalmente se presentan al usuario a través del dispositivo de salida, que por ejemplo puede ser la impresora o la pantalla de su Terminal. El mensaje recibido por el usuario ( $m'$ ) será normalmente una copia exacta del mensaje original ( $m$ ).

**1.1.3 Transmisión de datos analógicos y digitales.**<sup>3</sup> En la transmisión de datos desde una fuente a un destino, se debe tener en cuenta la naturaleza de los datos, cómo se propagan físicamente dichos datos, y qué procesamiento o ajustes se necesitarán a lo largo del camino para asegurar que los datos que se reciban sean inteligibles. Para todas estas consideraciones, el punto crucial es si se tratan de entidades digitales o analógicas. Los términos *analógico* y *digital* corresponden, en términos generales a *continuo* y *discreto*, respectivamente. Estos dos términos se aplican con frecuencia en las comunicaciones de datos a: datos, señalización, transmisión.

Se define dato como cualquier entidad capaz de transportar información. Las señales son representaciones eléctricas o electromagnéticas de los datos. La señalización es el hecho de la propagación física de las señales a través de un medio adecuado. Por último, se define transmisión como la comunicación de datos mediante la propagación y el procesamiento de señales.

---

<sup>3</sup>Ibid., p. 73.

**Transmisión.** Tanto las señales analógicas como las digitales se pueden transmitir a través del medio de transmisión adecuado, este determinará cómo se tratan estas señales. Las señales pueden representar datos analógicos (por ejemplo, voz) o datos digitales (por ejemplo, los datos binarios modulados en un modem).

En cualquier caso, la señal analógica se irá atenuándose (debilitando) con la distancia. Para conseguir distancias más largas, el sistema de transmisión analógico incluye amplificadores que inyectan energía en la señal. Desgraciadamente, el amplificador también inyecta energía en las componentes de ruido. Para conseguir distancias mayores, al utilizar amplificadores en cascada, la señal se distorsiona cada vez más. Para datos analógicos, como la voz, se puede tolerar una pequeña distorsión, ya que en ese caso los datos siguen siendo inteligibles. Sin embargo, para los datos digitales los amplificadores en cascada introducirán errores.

La transmisión digital, por contra, es dependiente del contenido de la señal. Una señal digital sólo puede transmitir a una distancia limitada, ya que la atenuación y otros aspectos negativos pueden afectar a la integridad de los datos transmitidos. Para conseguir distancias mayores se usan repetidores, un repetidor recibe la señal digital, regenera el patrón de ceros y unos y los retransmite. De esta manera se evita la atenuación.

Para señales analógicas se puede usar la misma técnica anterior si la señal transmitida transporta datos digitales. En este caso, el sistema de transmisión tendrá repetidores convenientemente espaciados en lugar de amplificadores. Dichos repetidores recuperan los datos digitales a partir de la señal analógica generan una señal analógica limpia. De esta manera el ruido no es acumulativo.

**1.1.4 Perturbaciones en la transmisión.**<sup>4</sup> En cualquier sistema de comunicaciones se debe aceptar que la señal que se recibe diferirá de la señal transmitida debido a varias adversidades y dificultades sufridas en la transmisión. En las señales analógicas, estas dificultades introducen alteraciones aleatorias que degradan la calidad de la señal. En las señales digitales, se producen bits erróneos: un 1 binario se transformará en un 0 y viceversa.

Las perturbaciones más significativas son:

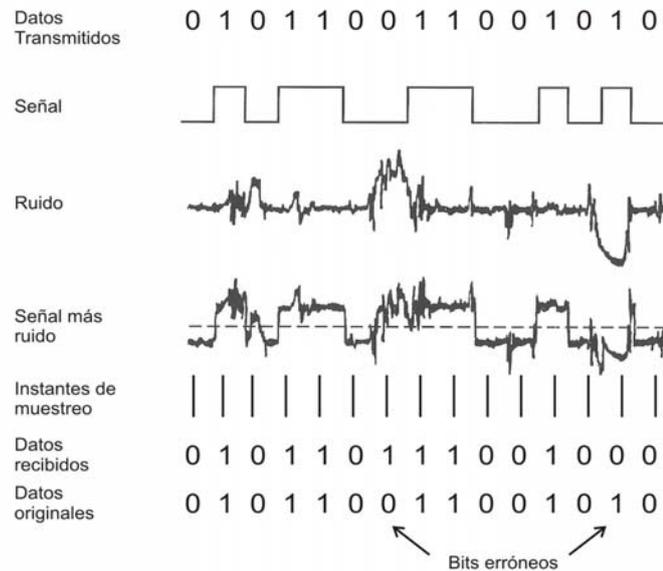
- La atenuación y el ruido.

En la figura 3, se observa el efecto del ruido, en la transmisión de una señal digital.

---

<sup>4</sup>TOMASI, Wayne. Sistema De Comunicaciones. 4 ed. México: Prentice Hall, 2003, p. 138.

Figura 3. Efecto del ruido en una señal digital



Fuente: TOMASI, Wayne. Sistema De Comunicaciones. 4 ed. México: Prentice Hall, 2003

**1.1.5 Ancho de banda (BW).** Es el rango de frecuencias en el que está contenida la mayor parte de la energía de la señal. Su unidad es el Hertz (hz).<sup>5</sup>

Estrictamente hablando, el ancho de banda de cualquier señal limitada en el tiempo es infinito. No obstante, en la práctica, la mayor parte de la potencia de la señal se concentra en una banda finita, y en ese caso, el ancho de banda consiste en la porción de espectro que contiene la mayor parte de la potencia.

La necesidad de renunciar a parte de la energía de la señal al limitar el ancho de banda a un determinado intervalo de frecuencias radica en el hecho de que señales con un espectro muy amplio de frecuencias no pueden transmitirse íntegramente por el medio de transmisión, puesto que estos suelen funcionar bien exclusivamente en un determinado rango de frecuencias. Por lo tanto, las frecuencias con menor energía se rechazan (se filtran). Un ejemplo muy claro de señal con espectro infinito es el pulso, base de las comunicaciones digitales.

Cuando los canales son analógicos se mide el ancho de banda en hertz, en cambio, cuando son digitales se habla de bps, Kbps, Mbps, etc. Si es un canal de voz, generalmente se habla de 3.1 kHz a 4 kHz, en canales digitales se habla de 64 Kbps

<sup>5</sup>Ibid., p. 73.

**1.1.6 Medios de transmisión.** El medio de transmisión constituye el soporte (físico o no) a través del cual emisor y receptor pueden comunicarse en un sistema de transmisión de datos. Se distinguen dos tipos de medios: *guiados* y *no guiados*. En ambos casos la transmisión se realiza por medio de ondas electromagnéticas. Los medios guiados conducen (guían) las ondas a través de un camino físico, ejemplos de estos medios son el cable coaxial, la fibra óptica y el par trenzado. Los medios no guiados proporcionan un soporte para que las ondas se transmitan, pero no las dirigen; como ejemplo de ellos se tienen el aire y el vacío.<sup>6</sup>

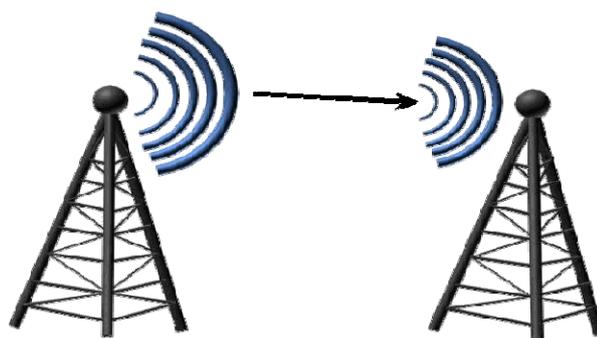
**Medios no guiados.** Cuando las ondas no están encauzadas (aire, agua, vacío): microondas, infrarrojos, radio. La radiocomunicación puede definirse como Telecomunicación realizada por medio de las ondas eléctricas. La Unión Internacional de Telecomunicaciones (UIT), define las ondas radioeléctricas como las ondas electromagnéticas que se propagan por el espacio sin guía artificial y cuyo límite superior de frecuencia se fija, en 3.000GHz.

Existen dos tipos fundamentales de transmisión inalámbrica:

**Direccionales:** La energía emitida se concentra en un haz, para lo cual se requiere que la antena receptora y transmisora estén alineadas. Cuanto mayor sea la frecuencia de transmisión, es más factible confinar la energía en una dirección.

**Omnidireccionales:** La antena transmisora emite en todas las direcciones espaciales y la receptora recibe igualmente en toda dirección.

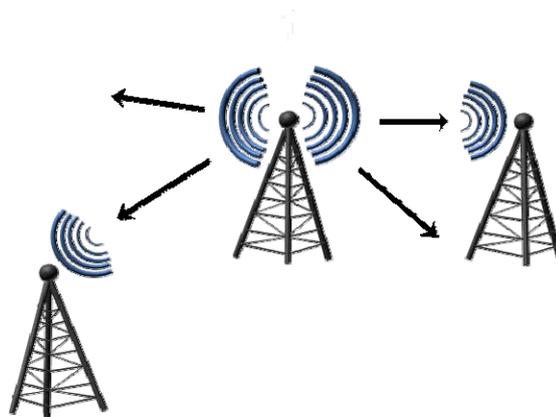
Figura 4. Comunicación direccional



Fuente: Ortiz, Hernán. Libro Electrónico Para La Asignatura De Telemática [CD ROM] [Pasto Nariño]: UDENAR, 2004.

<sup>6</sup>Ortiz, Hernán. Libro Electrónico Para La Asignatura De Telemática [CD ROM] [Pasto Nariño]: UDENAR, 2004. Medios de transmisión.

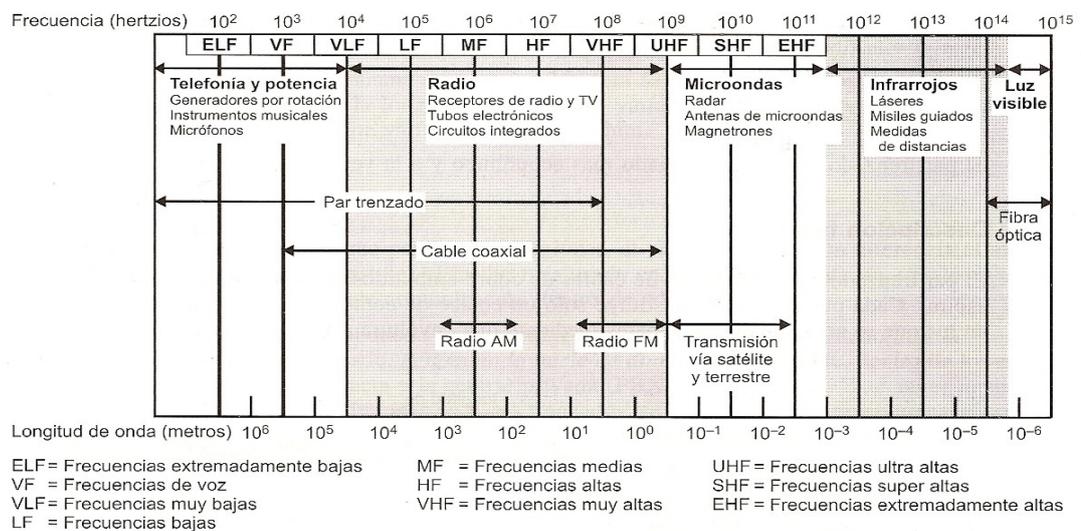
Figura 5. Comunicación omnidireccional



Fuente: Ortiz, Hernán. Libro Electrónico Para La Asignatura De Telemática [CD ROM] [Pasto Nariño]: UDENAR, 2004.

### 1.1.7 Espectro electromagnético.

Figura 6. Espectro electromagnético para las telecomunicaciones



Fuente: STALLINGS, William. Comunicaciones y Redes De Computadores. 6 ed. España: Prentice Hall, 2000.

En las bandas VLF, VF y MF, las ondas siguen la curvatura de la tierra. En las bandas HF y VHF, las ondas a nivel del suelo tienden a ser absorbidas por la tierra. Pero las que alcanzan la ionosfera, se refractan y se envían de regreso a nuestro planeta, en ciertas condiciones atmosféricas pueden rebotar varias veces.

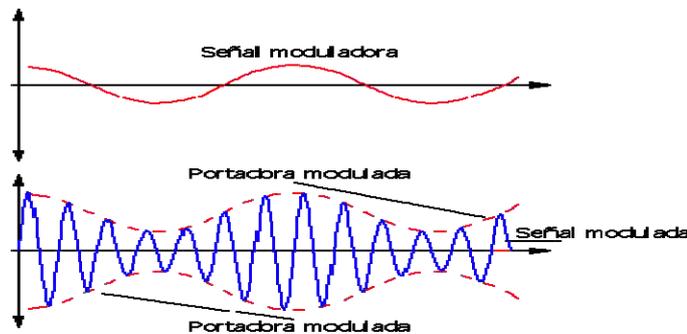
**1.1.8 Modulación.** Se denomina modulación, a la operación mediante la cual ciertas características de una onda denominada *portadora*, se modifican en función de otra denominada *moduladora*, que contiene información, a los efectos de ser transmitida.<sup>7</sup>

La onda en condiciones de ser transmitida, se denomina *señal modulada*.

*Señal Moduladora (señal de datos).* Es la que contiene la información.

*Señal Portadora.* Señal que aporta las características para adecuarla al medio de transmisión. *Señal Modulada.* Señal resultante con las características de la portadora y la información de la moduladora.

Figura 7. Modulación



Fuente: Ortiz, Hernán. Libro Electrónico para la asignatura de telemática [CD ROM] [Pasto Nariño]: UDENAR, 2004.

Tabla 1. Clases de modulación

PORTADORA MODULADORA	ANALÓGICA (Cos(x))		DIGITAL (Reloj)	
<b>ANALÓGICA</b>	AMPLITUD FRECUENCIA FASE	(AM) (FM) (PM)	AMPLITUD DURACIÓN POSICIÓN CODIFICACIÓN	(PAM) (PDM) (PPM) (PCM)
<b>DIGITAL</b>	AMPLITUD FRECUENCIA FASE	(ASK) (FSK) (PSK)	DECODIFICACIÓN	

Fuente: Ortiz, Hernán. Libro Electrónico para la asignatura de telemática [CD ROM] [Pasto Nariño]: UDENAR, 2004.

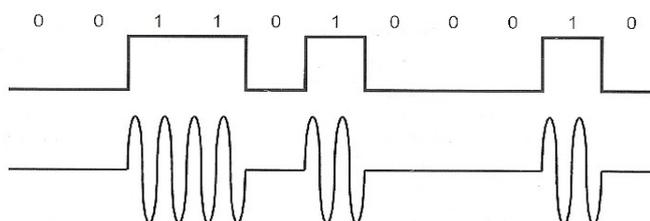
<sup>7</sup>Ortiz, Hernán. Libro Electrónico para la asignatura de telemática [CD ROM] [Pasto Nariño]: UDENAR, 2004. Modulación.

**1.1.9 Técnicas de modulación digital con portadora modulada.** Las señales de datos de terminales digitales y dispositivos afines, generalmente no se transmiten a gran distancia en la forma de señal de banda de base, es decir, tal como se generan, sino que se transmiten en forma de una señal modulada en forma analógica. En efecto, los impulsos o dígitos binarios modulan una portadora sinusoidal cuya frecuencia es compatible con el medio de transmisión utilizado; este tipo de transmisión se denomina “Transmisión Binaria mediante Portadora Modulada”.<sup>8</sup>

La modulación digital se realiza en un dispositivo denominado “MODEM” (de las Palabras **MO**dulador y **DE**Modulador) en el cual los dígitos binarios modulan la amplitud, la frecuencia o la fase de una señal sinusoidal, la portadora. Como la señal binaria de entrada es una secuencia de impulsos o dígitos binarios (de dos estados), las magnitudes moduladas tomarán también dos valores, de ahí que a estos procesos se los denomina también “Técnicas de Modulación Binaria”.

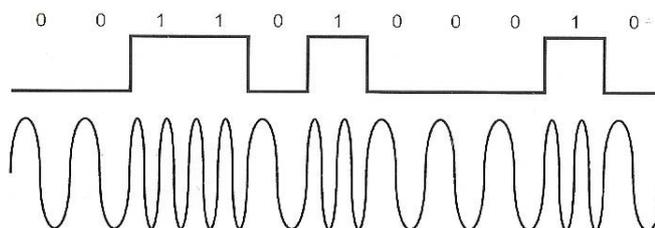
Las formas básicas de la modulación binaria mediante portadora modulada son: La Modulación Binaria de Amplitud (ASK), Modulación Binaria de Frecuencia (FSK) y Modulación Binaria de Fase (PSK).

Figura 8. Modulación ASK



Fuente: STALLINGS, William. Comunicaciones y Redes De Computadores. 6 ed. España: Prentice Hall, 2000.

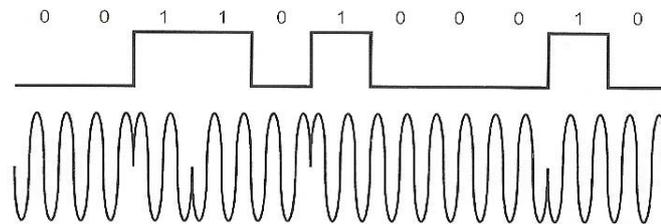
Figura 9. Modulación FSK



Fuente: STALLINGS, William. Comunicaciones y Redes De Computadores. 6 ed. España: Prentice Hall, 2000.

<sup>8</sup>TOMASI, Wayne. Sistema De Comunicaciones. 4ed. México: Prentice Hall, 2003, p. 217.

Figura 10. Modulación PSK



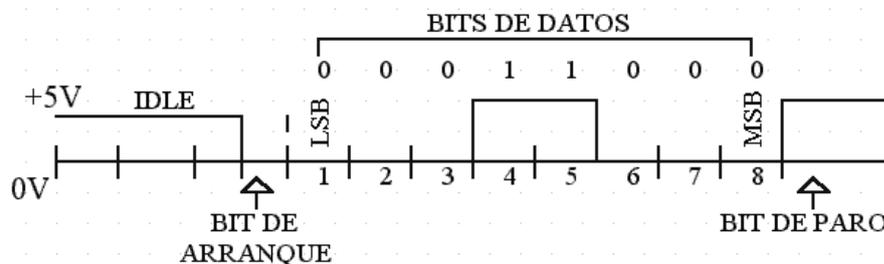
Fuente: STALLINGS, William. Comunicaciones y Redes De Computadores. 6 ed. España: Prentice Hall, 2000.

## 1.2 INTERFACES DE COMUNICACIÓN SERIAL.

Existen dos tipos de comunicaciones seriales: la síncrona y asíncrona. En la comunicación serial síncrona además de una línea sobre la cual se transmiten los datos, se necesita de una línea la cual contiene los pulsos de reloj que indican cuando un dato es valido. Ejemplos de este tipo de comunicación son: I2C, ONE WIRE y SPI.

En la comunicación serial asíncrona, no son necesarios los pulsos de reloj. La duración de cada bit esta determinada por la velocidad con la cual se realiza la transferencia de datos. Normalmente cuando no se realiza ninguna transferencia de datos, la línea del transmisor se encuentra en estado de (idle) este quiere decir en estado alto. Para iniciar la transmisión de datos, el transmisor coloca esta línea en bajo durante determinado tiempo, lo cual se le conoce como bit de arranque (start bit) y a continuación empieza a transmitir con un intervalo de tiempo los bits correspondientes al dato, empezando siempre por el BIT menos significativo (LSB), y terminando con el BIT mas significativo (MSB).

Figura 11. Estructura de un caracter que se trasmite en forma serial asíncrono



Fuente: LOPEZ, Eric. Tutorial del Protocolo RS-232. [Online]. España: Ingeniería en Microcontroladores.

Las tasas de transferencia de datos se miden en baudios, estos indican el número de veces que puede cambiar una señal en la línea de transmisión por segundo. En una interfaz serie, las señales cambian siempre a la misma frecuencia y se realiza una codificación binaria de la información de forma que cuando se quiere enviar un '1' se pone la línea a nivel alto y cuando se quiere enviar un '0' se pone la línea a nivel bajo. En este caso los baudios coinciden con el número de bits por segundo transferidos si se incluyen también los bits de comienzo, de fin y de paridad.<sup>9</sup>

Si el receptor no está sincronizado con el transmisor, este desconoce cuando se van a recibir los datos. Por lo tanto, el transmisor y el receptor deberán tener los mismos parámetros de velocidad, paridad, número de bits del dato transmitido y de BIT de parada.

**1.2.1 Norma RS-232.**<sup>10</sup> Ante la gran variedad de equipos, sistemas y protocolos que existen surgió la necesidad de un acuerdo que permitiera a los equipos de varios fabricantes comunicarse entre sí. La EIA (Electronics Industry Association) elaboró la norma RS-232, la cual define la interfase mecánica, los pines, las señales y los protocolos que debe cumplir la comunicación serial. Todas las normas RS-232 cumplen con los siguientes niveles de voltaje:

- Un "1" lógico es un voltaje comprendido entre  $-5\text{v}$  y  $-15\text{v}$  en el transmisor y entre  $-3\text{v}$  y  $-25\text{v}$  en el receptor.
- Un "0" lógico es un voltaje comprendido entre  $+5\text{v}$  y  $+15\text{v}$  en el transmisor y entre  $+3\text{v}$  y  $+25\text{v}$  en el receptor.

El envío de niveles lógicos (bits) a través de cables o líneas de transmisión necesita la conversión a voltajes apropiados. En los microcontroladores para representar un 0 lógico se trabaja con voltajes inferiores a  $0.8\text{v}$ , y para un 1 lógico con voltajes mayores a  $2.0\text{V}$ . En general cuando se trabaja con familias TTL y CMOS se asume que un "0" lógico es igual a cero voltios y un "1" lógico es igual a cinco voltios. La importancia de conocer esta norma, radica en los niveles de voltaje que maneja el puerto serial del ordenador, ya que son diferentes a los que utilizan los microcontroladores y los demás circuitos integrados. Por lo tanto, se necesita de una interfase que haga posible la conversión de los niveles de voltaje a los estándares manejados por los circuitos integrados TTL.

---

<sup>9</sup>LOPEZ, Eric. Tutorial Del Protocolo RS-232. [Online]. España: Ingeniería en Microcontroladores, enero del 2007. Disponible en Internet: <[wk3.iespana.es/Serial\\_RS232.pdf](http://wk3.iespana.es/Serial_RS232.pdf)>; p. 2.

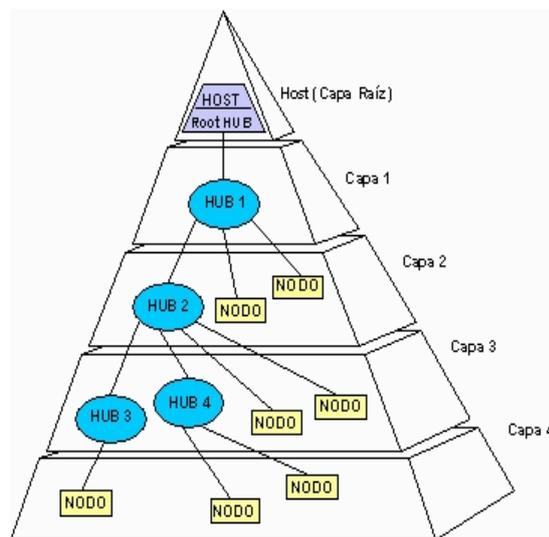
<sup>10</sup>Ibid., p. 5.

**1.2.2 El bus serie universal (USB).**<sup>11</sup> El USB es un estándar (1995) que define un bus utilizado para conectar periféricos al computador. La principal característica que tiene es que la conexión es muy sencilla, ya que utiliza un único conector para conectar a través de un bus serie todos los dispositivos. En él se definen los conectores y los cables, una topología especial tipo estrella para conectar hasta 127 dispositivos y protocolos que permiten la detección y configuración automática de los dispositivos conectados. USB 1.0 soporta dos tasas de transferencia diferentes, una baja de 1,5 Mbps para la conexión de dispositivos lentos de bajo costo (joysticks, ratones) y otra alta de hasta 12 Mbps para la conexión de dispositivos que requieren un mayor ancho de banda (discos o CDROMS).

A mediados del año 2000 aparece la versión 2.0 (USB 2.0) que multiplica la velocidad del bus por un factor de 30 o 40, llegando a alcanzar una velocidad de 480 Mbps, con una diferencia de costo casi inapreciable. Es compatible con la versión anterior y utiliza los mismos cables y conectores.

Este estándar define una topología de conexión en estrella, tal como se muestra en la siguiente figura, por medio de la incorporación de varios concentradores (*hubs*) conectados en serie. Cada concentrador se conecta por un lado al computador, que contiene una o dos interfaces de este tipo en la placa base, o a otro concentrador y, por otro lado, se conecta a varios dispositivos o incluso a otro concentrador.

Figura 12. Estructura de capas del bus USB



Fuente: Universal Serial Bus Revision 2.0 specification [Online]. [www.usb.org](http://www.usb.org).

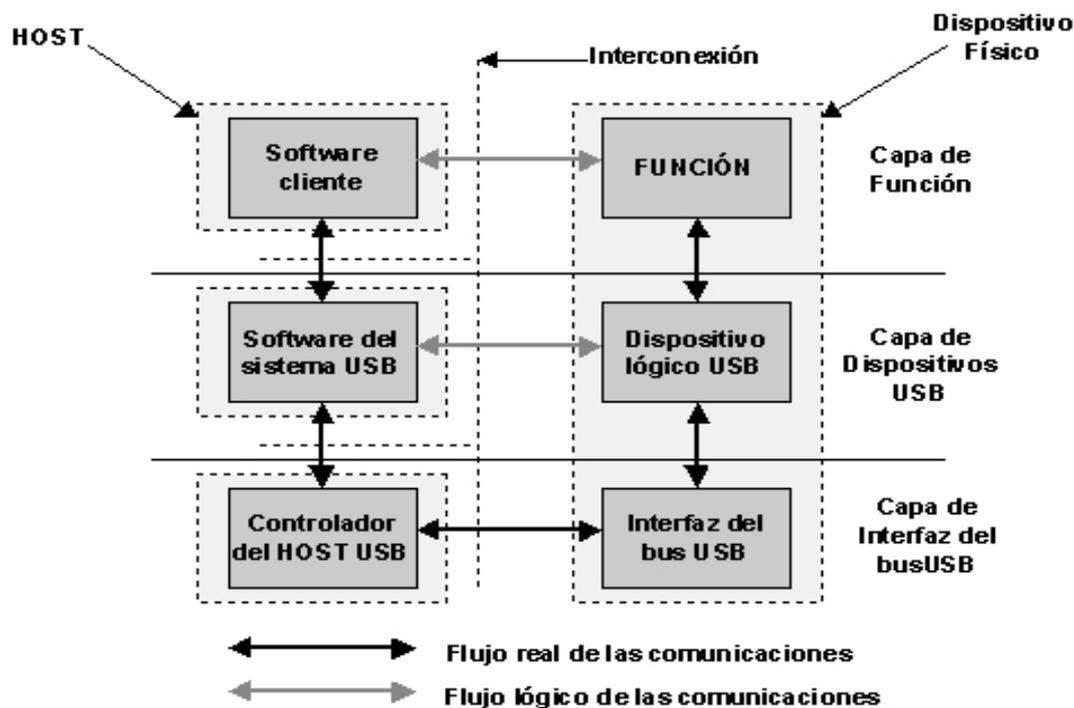
<sup>11</sup>Universal Serial Bus Revision 2.0 specification [Online]. [www.usb.org](http://www.usb.org). Revisión 2.0 Abril 27, 2000. Disponible en Internet: [http://www.usb.org/develo pers/docs/usb\\_20.pdf](http://www.usb.org/develo pers/docs/usb_20.pdf)

Los hubs tienen la misión de ampliar el número de dispositivos que se pueden conectar al bus. Son concentradores cableados que permiten la conexión simultánea de múltiples dispositivos y lo más importante es que se pueden concatenar entre sí ampliando la cantidad de puertos disponibles para los periféricos. El concentrador detecta cuándo un periférico es conectado o desconectado a/de uno de sus puertos, notificándolo de inmediato al controlador de USB. También realiza funciones de acoplamiento de las velocidades de los dispositivos más lentos.

La topología del bus USB, se puede dividir en tres partes.

- *La capa física.* Como están conectados los elementos físicamente.
- *La capa lógica.* Los roles y las responsabilidades de los elementos USB.
- *La relación software del cliente-función.* Como se ven mutuamente el software del cliente y las interfaces de las funciones relacionadas.

Figura 13. Flujo de comunicaciones entre dispositivo USB y el host



Fuente: Universal Serial Bus Revision 2.0 specification [Online]. [www.usb.org](http://www.usb.org).

El flujo de datos del bus USB desde un punto de vista lógico hay que entenderlo como una serie de puntos finales (*endpoints*), que se agrupan en conjuntos que dan lugar a interfaces, las cuales permiten controlar la función del dispositivo.

La arquitectura se puede dividir en tres niveles o capas. En el nivel más bajo el controlador de host USB se comunica con la interfaz del bus utilizando el cable USB, mientras que en un nivel superior el software USB del sistema se comunica con el dispositivo lógico utilizando la tubería de control por defecto. En lo que al nivel de función se refiere, el software cliente establece la comunicación con las interfaces de la función a través de tuberías asociadas a puntos finales.

Cada dispositivo USB está compuesto por unos puntos finales independientes y una dirección única asignada por el sistema en tiempo de conexión, de forma dinámica. A su vez, cada punto final dispone de un identificador único dentro del dispositivo (número de endpoint) que viene asignado de fábrica, además de una determinada orientación del flujo de datos. Cada punto final es por si solo una conexión simple, que soporta un flujo de datos de entrada o de salida.

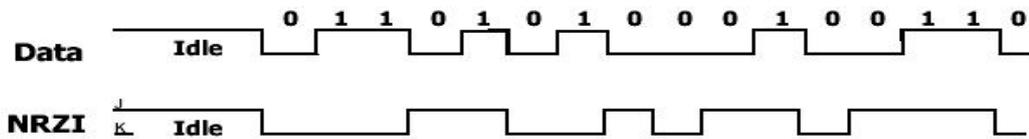
Una tubería USB es una conexión lógica entre un punto final del periférico y el software del host, que permite intercambiar datos entre ellos. La tubería que esta formada por el punto final 0 se denomina tubería de control por defecto. Esta tubería está siempre disponible una vez se ha conectado el dispositivo y ha recibido un reset del bus. El resto de tuberías aparecen después de que se configure el dispositivo. La tubería de control por defecto es utilizada por el software USB del sistema para obtener la identificación y para configurar al periférico.

Existe una gran variedad de dispositivos USB que se conectan todos al mismo bus. La característica más importante es que todos ellos utilizan el mismo tipo de cable y de conector, se conectan de la misma forma. El host decide qué dispositivo puede acceder al bus, utilizando un protocolo parecido al de paso de testigo. Este protocolo se caracteriza porque entre los diferentes dispositivos se va pasando un identificador a lo largo del tiempo que permite la utilización del bus. El host USB tiene las funciones de:

- Detectar la conexión/desconexión de dispositivos y configurarlos.
- Controlar las transferencias de datos y de control que tienen lugar en el bus.
- Realización de auditorías sobre la actividad del sistema.
- Servir como fuente de alimentación a los dispositivos.

El USB define dos líneas para transmitir datos y otras dos para transmitir potencia. Los datos se transmiten de forma balanceada a velocidades entre 1,5 Mbps y 12 Mbps. La señal se envía codificada en un código auto reloj de no retorno a cero invertido (NRZI) para poder incluir junto con los datos información de sincronización.

Figura 14. Codificación NRZI



Fuente: Universal Serial Bus Revision 2.0 specification [Online]. [www.usb.org](http://www.usb.org).

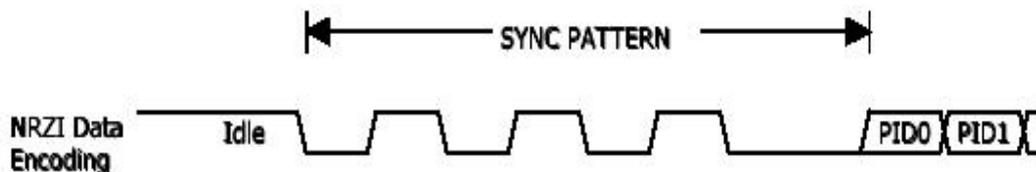
Debido a que cadenas de unos pueden producir largos periodos en los que la señal no cambia dando lugar a problemas de sincronización, se introducen los bits de relleno. Cada 6 bits consecutivos a "1" se inserta un bit a "0" para forzar un cambio, de esta forma el receptor puede volverse a sincronizar. El relleno bits empieza con el patrón de señal Sync. El "1" que finaliza el patrón de señal Sync es el primer uno en la posible primera secuencia de seis unos.

En las señales a velocidad media o baja, el relleno de bits se utiliza a lo largo de todo el paquete sin excepción. De modo que un paquete con siete unos consecutivos será considerado un error y por lo tanto ignorado.

En el caso de la velocidad alta se aplica el relleno de bits a lo largo del paquete, con la excepción de los bits intencionados de error usados en EOP a velocidad alta.

- **Sync:** Teniendo en cuenta que 0 y 1 representan respectivamente nivel bajo y nivel alto, el patrón de señal Sync emitido, con los datos codificados, es de tres pares 01 seguidos de dos 0 para el caso de velocidad media y baja. Para velocidad alta es una secuencia de 15 pares 01 seguidos de dos 0. A continuación se muestra la figura para el caso de velocidad media y baja:

Figura 15. Señal sync de sincronismo para baja velocidad



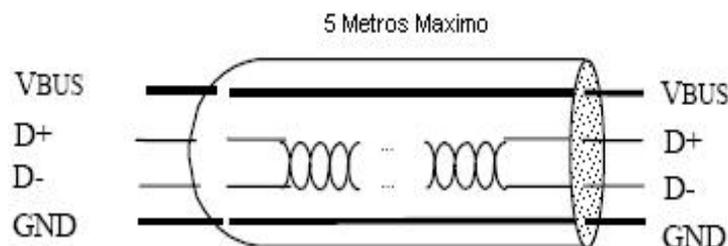
Fuente: Universal Serial Bus Revision 2.0 specification [Online]. [www.usb.org](http://www.usb.org).

El patrón de señal Sync siempre precede al envío de cualquier paquete, teniendo como objetivo que el emisor y el receptor se sincronicen y se preparen para emitir y recibir datos respectivamente. Se parte de que el estado de reposo de la señal es 1, se puede interpretar Sync como una secuencia impar de "0's" y un "1" que se inserta antes de los datos.

- **EOP ("End Of Packet")**. A todo paquete le sigue EOP, cuya finalidad es indicar el final del paquete. En el caso de velocidad media y baja el EOP consiste en que, después del último bit de datos en el cual la señal estará o bien en estado alto, o bien en estado bajo, se pasa al estado SE0 durante el periodo que se corresponde con el ocupado por dos bits, finalmente se transita al estado alto que se mantiene durante 1 bit. Esta última transición indica el final del paquete. En el caso de la velocidad alta se utilizan bits de relleno erróneos, que no están en el lugar correcto, para indicar el EOP. Concretamente, el EOP sin aplicar codificación consistiría en añadir al final de los datos la secuencia 0111 1111.

Las líneas de alimentación (Vbus y GND) evitan la necesidad de utilizar fuentes de alimentación externas. Tiene una tensión de 5V y la corriente se limita a un máximo de 500 miliamperios por razones de seguridad, siendo el consumo y la configuración eléctrica totalmente transparente al usuario. La distancia entre dos periféricos conectados al mismo cable no debe ser superior a 5 metros para evitar problemas de caídas de tensión.

Figura 16. Esquema del cable USB



Fuente: Universal Serial Bus Revision 2.0 specification [Online]. [www.usb.org](http://www.usb.org).

El computador identifica automáticamente el dispositivo que se conecta mientras opera y lo configura sin tener que instalar drivers específicos del fabricante. Al comienzo se detectan los dispositivos conectados midiendo los niveles de voltaje de las líneas. Si un dispositivo está conectado, entonces envía información sobre el tipo o la clase a la que pertenece, qué modo de transferencia utilizará y cuáles son sus necesidades de ancho de banda. El host reconoce el dispositivo buscando en la lista de drivers del sistema operativo y teniendo en cuenta los demás dispositivos conectados, asigna un ancho de banda determinado. De la misma forma también se puede desconectar los dispositivos del sistema. El controlador USB del host asigna un número diferente de dispositivo a cada uno de los periféricos que se conectan a este bus. Para empezar la transferencia, éste envía un paquete que identifica al dispositivo objeto de la transferencia. El protocolo soporta cuatro tipos de transferencias.

- **Control.** Son transferencias que se utilizan para leer información de los descriptores en los registros de los dispositivos (llamados *endpoints*), interpretarla y poder configurarlos.
- **Interrupción.** Usadas en los periféricos del tipo de los controladores de juegos, teclados y ratones, cuya comunicación es unidireccional y poco frecuente.
- **Masiva.** Son transferencias no periódicas que precisan de todo el ancho de banda disponible. Utilizadas por las impresoras y los scanners.
- **Isócrona.** Dedicadas a las transferencias de telecomunicaciones, como voz o vídeo, que garantiza unas tasas de transferencia constantes. Se caracterizan porque el número de pulsos de reloj que transcurren entre la transmisión de dos caracteres es constante, por lo tanto, se está enviando información constantemente entre el host y el dispositivo.

### 1.3 MICROCONTROLADORES

Un microcontrolador es un circuito integrado programable que contiene todos los componentes de un computador. Se utiliza para controlar el funcionamiento de una tarea determinada, debido a su reducida medida suele ir incorporado en el propio dispositivo que gobierna. Esta última característica es la que le confiere la denominación de controlador incrustado (*embedded controller*).<sup>12</sup>

El microcontrolador es un computador dedicado. En su memoria solamente reside un programa destinado a gobernar una aplicación determinada; sus líneas de entrada/salida soportan la conexión 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.

Un microcontrolador posee todos los componentes de un computador pero con unas características fijas que no pueden alterarse. Todos disponen de los bloques esenciales: procesador, memoria de datos y de instrucciones, módulos de E/S, oscilador de reloj y módulos controladores de periféricos. Además de estos elementos, existen una serie de recursos especiales que los fabricantes pueden ofertar, algunos amplían las capacidades de las memorias, otros incorporan nuevos recursos y hay quienes reducen las prestaciones al mínimo para aplicaciones muy simples. Depende del programador el encontrar el modelo mínimo que se ajuste a sus requerimientos y así minimizar el costo, el hardware y el software. Algunos de los principales recursos específicos que incorporan los microcontroladores son:

<sup>12</sup>DUQUE, Edison. Curso Básico De Microcontroladores PIC. Colombia: CEKIT S.A. 1997. p.5.

- Temporizadores (*Timers*).
- Perro guardián (*Watchdog*).
- Protección frente a fallo de alimentación (*Brown-out*).
- Estado de bajo consumo.
- Conversores AD y DA.
- Modulador de anchura de pulsos PWM.
- Comparadores analógicos.
- Puertos de E/S digital.
- Puertos de comunicación: serie, CAN, USB, I2C.

Las principales ventajas que se pueden encontrar en el uso de microcontroladores son:

- Gestión eficiente de procesos.
- Aumento de la fiabilidad.
- Reducción del tamaño, consumo y costo.
- Mayor flexibilidad (únicamente se requiere la reprogramación).

Existe una gran diversidad de microcontroladores. Quizá la clasificación más importante sea la que se refiere al número de bits a manejar y así se los conoce como microcontroladores de 4, 8, 16 o 32 bits. Aunque las prestaciones de los microcontroladores de 16 y 32 bits son superiores a los de 4 y 8 bits, la realidad es que los microcontroladores de 8 bits dominan el mercado y los de 4 bits se resisten a desaparecer. La razón de esta tendencia es que los microcontroladores de 4 y 8 bits son apropiados para la gran mayoría de las aplicaciones, lo que hace innecesario emplear microcontroladores más potentes y consecuentemente más caros.

En cuanto a las técnicas de fabricación, prácticamente la totalidad de los microcontroladores actuales se fabrican con tecnología CMOS (*Complementary Metal Oxide Semiconductor*). Esta tecnología supera a las técnicas anteriores por su bajo consumo y alta inmunidad al ruido.

## 2. DESARROLLO

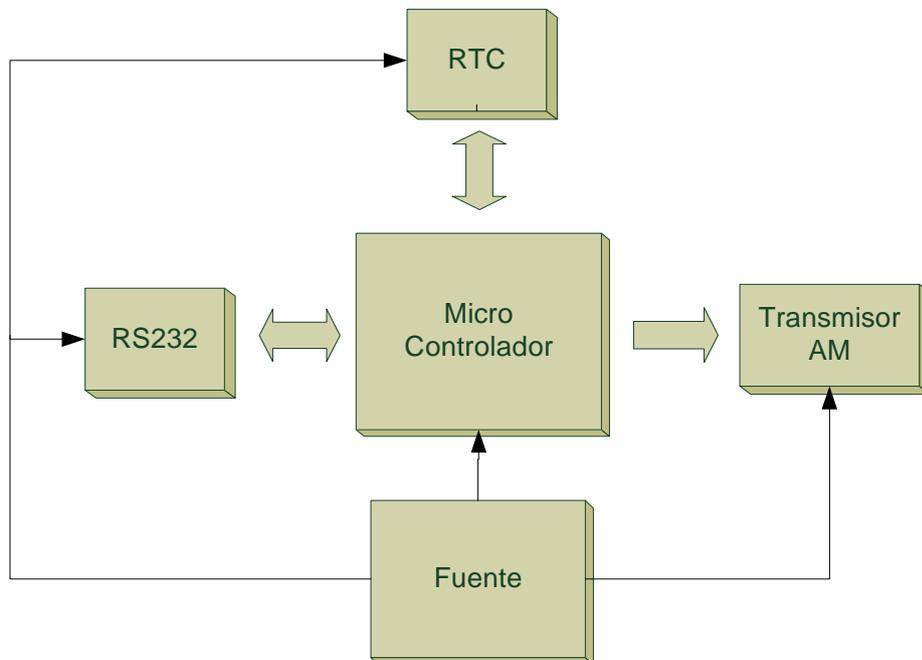
### 2.1 DESCRIPCIÓN GENERAL

*El sistema de control de tiempo del transporte de buses urbanos está conformado por tres módulos independientes módulo fijo, módulo móvil y módulo de configuración, que interactúan entre si para lograr el objetivo de automatizar el proceso. Cada uno de los módulos es constituido por subsistemas que cumplen ciertas características mínimas para lograr el funcionamiento del conjunto.*

Para que los módulos fijo y móvil puedan interactuar entre si, se incorpora al sistema una unidad de comunicación por radio frecuencia para que permitan el intercambio inalámbrico de datos entre ellos.

**2.1.1 Módulo fijo.** Se encuentra programado para realizar la transmisión de datos (hora, fecha y lugar donde se encuentra ubicado este elemento), esta función se realiza vía RF (radio Frecuencia) en modulación ASK.

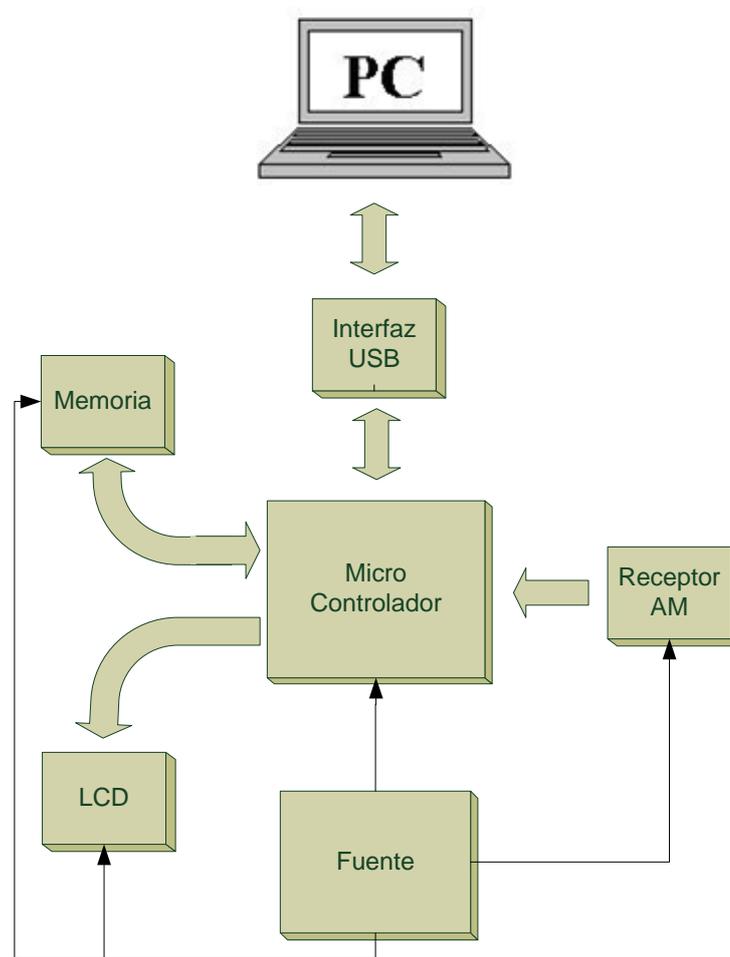
Figura 17. Diagrama de bloques del módulo fijo



El hardware del dispositivo esta conformado por un transmisor ASK TLP434A, un microcontrolador PIC16F628A, un RTC (real time clock) DS1307, una interfaz RS232 y la etapa de alimentación.

**2.1.2 Módulo móvil.** Su función es la de recibir la información que envía el módulo fijo, los procesa y almacena en su memoria, la recepción se realiza por RF. Para la visualización de los datos recolectados se capturan desde un computador mediante interfaz USB.

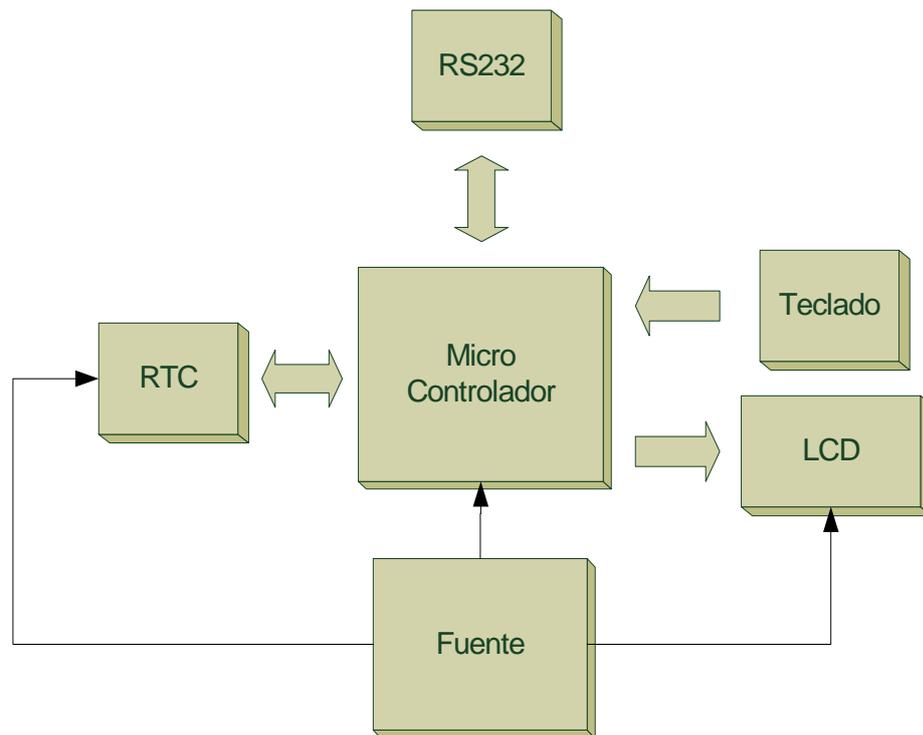
Figura 18. Diagrama de bloques del módulo móvil



Se constituye por receptor ASK RLP434B, un microcontrolador PIC18F2550, una memoria EEPROM 24LS256, una pantalla LCD 2x16, interfaz USB y el módulo de alimentación.

**2.1.3 Módulo de configuración.** Su tarea es sincronizar los RTC de los módulos fijos y además asignarle el lugar de control, para ello se cuenta con una interfaz RS232 en los dos dispositivos.

Figura 19. Diagrama de bloques del módulo de configuración



Consta de un microcontrolador PIC18F2550, un RTC (real time clock) DS1307, una pantalla LCD 2x16, un teclado matricial 3x4, una interfaz RS232, tres pulsadores de control y el módulo de alimentación.

## 2.2 DESCRIPCIÓN DEL HARDWARE.

**2.2.1 Selección de microcontroladores.** Los microcontroladores elegidos para el desarrollo del proyecto son de la empresa Microchip y pertenecen a la familia PIC de la serie 16 y 18 son dispositivos de bajo consumo y altas prestaciones como el de la serie 18 que cuenta con la posibilidad de realizar comunicación USB.

Para la programación del firmware de los Microcontroladores PIC se opta por utilizar el extendido compilador CCS Compiler, este incorpora bibliotecas que permiten agilizar y mejorar el desarrollo de proyectos.

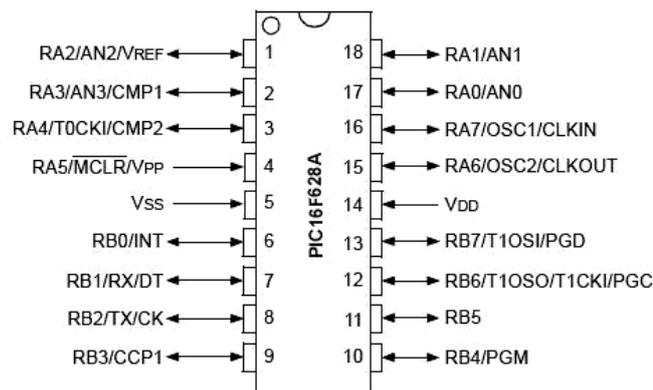
**2.2.1.1 Microcontrolador dispositivo fijo y unidad de alimentación.** El módulo fijo es el encargado de enviar la información para establecer la ubicación y el tiempo, la unidad central de proceso se basa en el microcontrolador PIC16F628A<sup>13</sup>, en el cual se programan rutinas para la transmisión de datos al módulo móvil y a partir de estos se automatiza el sistema de control de tiempo. Las principales características del microcontrolador son:

Tabla 2. Especificaciones microcontrolador PIC16F628A

Características	Detalles
Rango de Voltaje	2 a 5.5 VDC
Frecuencia de Operación	DC a 20 Mhz
Memoria Flash de Programa	2 Kbytes
Memoria RAM de datos	224 Bytes
Memoria EEPROM	128 Bytes
Puertos de Entrada/salida	2(A,B)
Fuentes de Interrupción	10
Instrucciones	35
Protección de código	Programable
Oscilador Interno	48 Khz a 4 Mhz
Comparadores	2
Comunicación Seriales	USART, I <sup>2</sup> C

En la siguiente figura, se puede observar el diagrama de pines de entrada y salida del PIC16F628A.

Figura 20. Encapsulado PIC16F628A



Fuente: PIC16F628A datasheet [Online]. Microchip Technology Inc.

<sup>13</sup>PIC16F628A datasheet [Online]. Microchip Technology Inc. 2005. Disponible en Internet: <[http://www.microchip.com/stellent/idcplg?IdcService=SS\\_GET\\_PAGE&nodeId=1335&dDocName=en010210](http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1335&dDocName=en010210)>.

La primera letra que hay en los diferentes modelos de PIC, en este caso la F (16F628A), indica el tipo de memoria que posee, en este caso memoria tipo Flash, que es una memoria no volátil, de bajo consumo de corriente que se puede borrar y escribir eléctricamente. Tiene igual funcionamiento que las memorias RAM o ROM pero es más veloz, su consumo energético es mínimo y tiene menor tamaño, a diferencia de la ROM la memoria Flash es programable en el mismo circuito, es más rápida y de mayor densidad que la EEPROM.

La fuente de alimentación o fuente de conmutación de voltaje la cual es la encargada de generar voltaje DC de 5V para alimentar cada uno de los dispositivos, además permite la conmutación automática a un banco de baterías de 4.8V DC de respaldo en ausencia de la energía principal, el microcontrolador en este circuito es el encargado de realizar el monitoreo del estado de carga de la batería y generar pulsos de 120Hz por un tiempo establecido para su recarga. El microcontrolador utilizado para la realización de esta tarea es el PIC16F628A ya que cuenta con comparadores de señal analoga que se utilizarán en la fuente para medir el nivel carga de la batería.

**2.2.1.2 Microcontrolador dispositivos móvil y de configuración.** El módulo móvil es el encargado de recibir los datos de hora, fecha y lugar que son transmitidos por el módulo fijo, además realiza la corrección de errores procesa los datos y los almacena en una memoria EEPROM externa, también permite establecer la comunicación serial USB con el computador, para la recolección y visualización de la información, por medio de una aplicación desarrollada en el programa Visual Basic. La unidad central de proceso se basa en el microcontrolador de gama media PIC18F2550<sup>14</sup>, en él se programa rutinas para la recepción, almacenamiento externo y recopilación de estos al PC.

Para la selección del PIC, se tuvo en cuenta los siguientes requerimientos según el orden de prioridad.

1. Disponer de un puerto de comunicación USB.
2. Longitud de palabra de 8 bits.

Teniendo en cuenta lo anterior, se optó por el PIC18F2550, que dispone de un puerto USB que puede funcionar a baja velocidad (1.5Mbps) como a alta velocidad (12Mbps). El módulo de configuración el cual es el encargado de actualizar los datos al reloj del dispositivo fijo por comunicación serial cuenta también en su unidad central de proceso con el PIC18F2550, en este se programan rutinas de

<sup>14</sup>PIC18F2550 datasheet [Online]. Microchip Technology Inc. 2004. Disponible en Internet: <[http://www.microchip.com/stellent/idcplg?IdcService=SS\\_GET\\_PAGE&nodeId=1335&dDocName=en010280](http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1335&dDocName=en010280)>.

interfaz gráfica, lectura y escritura de RTC interno, transmisión y recepción de datos por protocolo RS232.

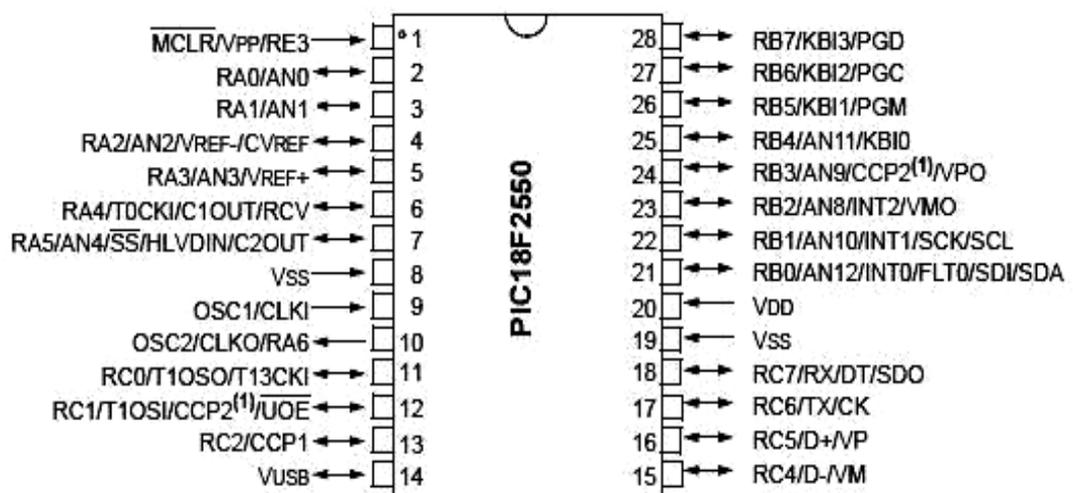
La siguiente tabla, muestra las características del microcontrolador.

Tabla 3. Especificaciones microcontrolador PIC18F2550

Características	Detalles
Rango de Voltaje	2 a 5.5 VDC
Frecuencia de Operación	DC a 48 Mhz
Memoria Flash de Programa	32 Kbytes
Memoria SRAM de datos	2 Kbytes
Memoria EEPROM	256 Bytes
Puertos de Entrada/salida	4(A,B,C,E)
Fuentes de Interrupción	19
Instrucciones	83
Protección de código	Programable
Oscilador Interno	8 (38 Khz a 8 Mhz)
comunicación Seriales	USART, MSSP
Módulo USB	1

En la siguiente figura, se observa el encapsulado del PIC18F2550 con sus pines de entrada y salida.

Figura 21. Encapsulado PIC18F2550



Fuente: PIC18F2550 datasheet [Online]. Microchip Technology Inc.

**2.2.2 Selección módulos de radio frecuencia.** La etapa de comunicación por radio frecuencia es la más relevante dentro del proyecto, ya que ésta permite que el registro de los datos de tiempo sea automático.

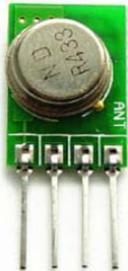
Con el fin de establecer la comunicación punto a punto entre la estación fija y la estación móvil, se requiere dos módulos de radio frecuencia, los cuales mediante un proceso de configuración y corrección de errores establecen la comunicación para el intercambio de datos.

Para el uso de frecuencias del espectro electromagnético, se debe tener en cuenta la regulación que hace el Ministerio de Telecomunicaciones de estas y la asignación de bandas de frecuencias radioeléctricas para su libre utilización dentro del territorio nacional.

La *Resolución número 000797 del 8 de junio del 2001*<sup>15</sup> emanada por el Ministerio de Telecomunicaciones se hace la asignación frecuencias radioeléctricas para su uso libre por parte del público en general y define las características técnicas de operación para su uso.

**2.2.2.1 Módulo de transmisión.** El módulo utilizado para realizar la comunicación inalámbrica del módulo fijo al módulo móvil es el transmisor miniatura RF TLP434A (ultra small transmitter)<sup>16</sup> sus principales características son.

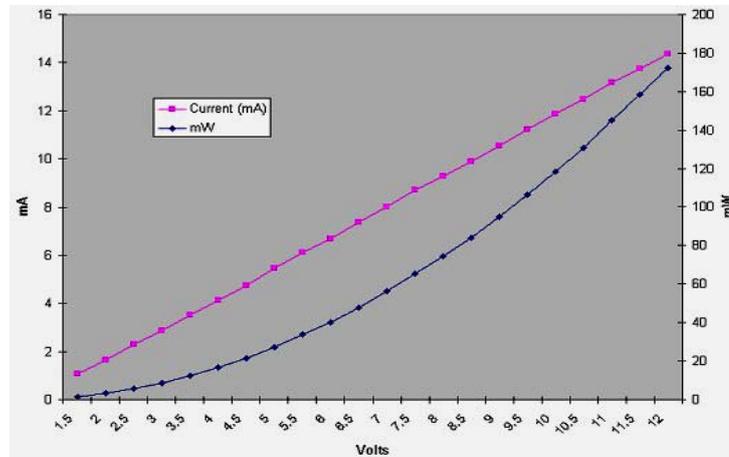
Tabla 4. Características principales del transmisor TLP434A

	Modulación	ASK
	Voltaje de operación	3 - 12 VDC
	Frecuencia de operación	433.92 Mhz
	Alcance Promedio	100 mts
	Temperatura de trabajo	-20 a +85 °C
	Rata de datos	8 Kbps
	Corriente de Input (3V)	4.9 mA
	dimensiones	Alto(13.3mm), Ancho(10,3mm)

<sup>15</sup>Resolución Número 000797 De 8 De Junio De 2001 [Online]. Ministerio De Comunicaciones. Colombia: 2001. Disponible en Internet: <[http://www.mincomunicaciones.gov.co/mincom/src/?page=../mods/legislacion/legislacion\\_user&id=271&state=V&id\\_tool=0](http://www.mincomunicaciones.gov.co/mincom/src/?page=../mods/legislacion/legislacion_user&id=271&state=V&id_tool=0)>.

<sup>16</sup>TLP434A datasheet [Online]. Laipac Technology Inc. 2001. Disponible en Internet: <[http://www.laipac.com/easy\\_434\\_eng.htm](http://www.laipac.com/easy_434_eng.htm)>.

Figura 22. Grafico corriente vs voltaje y potencia vs voltaje



Fuente: TLP434A Power output [Online]. Laipac Technology Inc.

La frecuencia de operación del transmisor esta centrada a 443.92 Mhz que se encuentra dentro de las frecuencias de uso civil de acuerdo con siguiente tabla, tomada de la resolución 000797 del Ministerio de Comunicaciones.

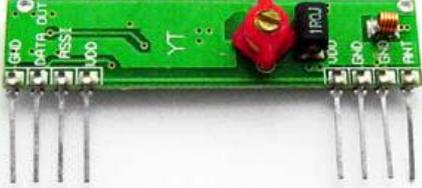
Tabla 5. Bandas de frecuencias para aplicaciones de telemetría y telecontrol con bajos niveles de potencia

FRECUENCIAS (MHz)	LÍMITE DE POTENCIA O DE INTENSIDAD DE CAMPO	APLICACIÓN	
0,045 a 0,490	1 mW	Localizadores de cables	
0,535 a 1,705	100 mW	Telemetría Biomédica	
26,957 a 27,283	300 mW	Controles remotos para modelos	
29,72 a 30,0	300 mW		
36,0 a 36,6	300 mW		
72,0 a 74,8	300 mW		
174 a 216	700 nW		Telemetría Biomédica
433 a 434,79	10 mW	Telecomando, Telecontrol, Controles remotos para modelos	
433,0 a 434,79	1 mW	Medidores de agua	
451,025 a 451,675			
426,0250 a 426,1375	1 mW	Telemetría, Telecontrol	
426,0375 a 426,1125	1 mW		
429,2500 a 429,2375	10 mW		
429,8125 a 429,9250	10 mW		
433,0 a 434,79	10 mW		
449,8375 a 469,9250	10 mW		
469,4375 a 469,4875	10 mW		
894 a 896	500 uV /m (a 3 m)		Medición de características de materiales
897,125 a 897,500	500 uV /m (a 3 m)		
905 a 908	500 uV /m (a 3 m)		
915 a 924	500 uV /m (a 3 m)		
924 a 928	500 uV /m (a 3 m)		
928 a 929	500 uV /m (a 3 m)		
932 a 935	500 uV /m (a 3 m)		
936,125 - 940,000	500 uV /m (a 3 m)		

Fuente: Resolución Número 000797 De 8 De Junio De 2001 [Online]. Ministerio De Comunicaciones. Colombia: 2001.

**2.2.2.2 Módulo de recepción.** El módulo encargado de recibir los datos enviados por el módulo fijo es el receptor miniatura RLP434A<sup>17</sup>. Sus principales características son:

Tabla 6. Características principales del receptor RLP434A

	Demodulación	ASK
	Voltaje de operación	3.3 - 6 VDC
	Frecuencia de sintonización con filtro SAW	433.92 Mhz
	Salida	Serial
	Rata de datos	4.8 Kbps
	Corriente de Operación	4.5 mA
	Compatible con Transmisor	TLP434A
	dimensiones	Alto(11.5mm), Ancho(43.4mm)

La antena es un dispositivo que sirve para transmitir y recibir ondas de radio, convierte la señal serial de salida de transmisor en ondas electromagnéticas, que se pueden transmitir por el espacio libre y además capta las ondas RF en el receptor. A continuación se observa las características de la antena WLP434<sup>18</sup> utilizada en el módulo móvil.

Tabla 7. Características principales Antena WLP-434 RF

	Frecuencia de Resonancia	434 +/-10 Mhz
	Radiación	Omnidireccional
	Polarización	Vertical
	Impedancia	50 Ohm

<sup>17</sup>RLP434A datasheet [Online]. Laipac Technology Inc. 2001. Disponible en Internet: < [http://www.laipac.com/easy\\_434\\_eng.htm](http://www.laipac.com/easy_434_eng.htm) >.

<sup>18</sup>WLP434 RF & Cordless Phone Antenna [Online]. Laipac Technology Inc. 2001. Disponible en Internet: < [http://www.laipac.com/ant\\_WLP\\_434\\_eng.htm](http://www.laipac.com/ant_WLP_434_eng.htm) >.

**2.2.3 RTC (Reloj de Tiempo Real).** La función principal del RTC es ser el reloj calendario de los módulos de configuración y fijo.

Para RTC se emplea el circuito integrado DS1307,<sup>19</sup> en presentación DIP de 8 pines, de Dallas semiconductor, es un reloj/calendario que cuenta los segundos, minutos, horas, día de la semana, mes, día del mes y el año. El fin de mes es ajustado automáticamente para aquellos meses que tienen menos de 31 días, e incluye compensación por año bisiesto. Es válido hasta el año 2100, el reloj opera entre los formatos de 24 horas o 12 horas con indicador de AM/PM. Al presentarse fallas en la alimentación (VDC) el DS1307 cambia automáticamente al modo de operación con batería de emergencia para no perder su tiempo y la fecha, mientras se mantenga apagada la alimentación principal, su consumo en esta condición es menor a 500nA. El DS1307 utiliza un oscilador, basado en un cristal de cuarzo externo de 32.768KHz., para marcar el paso cada segundo y actualizar los datos en sus registros internos. Este dispositivo utiliza el protocolo de comunicación serie síncrona I<sup>2</sup>C, para comunicar al RTC con un microcontrolador, ya sea para leer o escribir los datos que almacena o bien configurar la operación del circuito integrado.

Figura 23. Distribución de pines del RTC DS1307



Fuente: DS1307 Serial Real -Time Clock datasheet [Online]. Dallas Semiconductor.

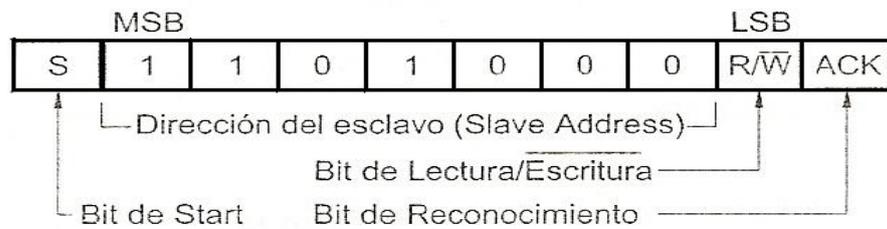
Tabla 8. Características principales del RTC DS1307

Características	Detalle
Rango de Voltaje	4 a 5.5VDC
Oscilador Externo	32.768Khz
comunicación Seriales	I <sup>2</sup> C
Frecuencia de Reloj SCL	0 a 100Khz
Batería de Backup	3 VDC
Corriente de Operación	1.5mA

<sup>19</sup>DS1307 Serial Real -Time Clock datasheet [Online]. Dallas Semiconductor. 2000. Disponible en Internet: < [http://www.maxim-ic.com/quick\\_view2.cfm/qv\\_pk/2688](http://www.maxim-ic.com/quick_view2.cfm/qv_pk/2688)>.

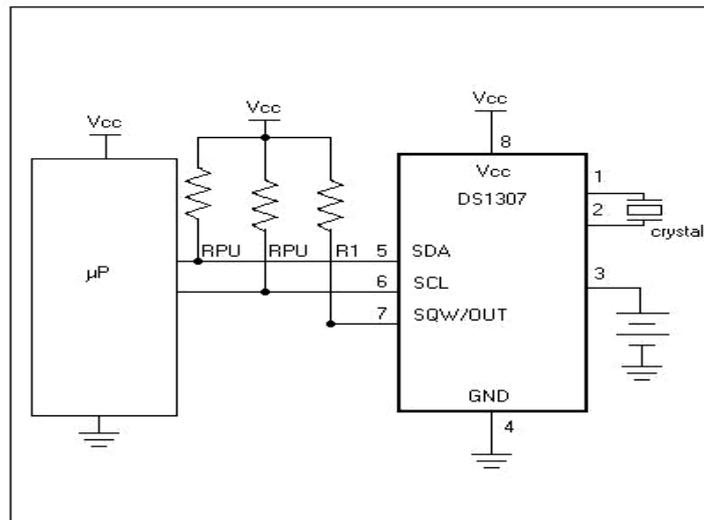
En la implementación de la comunicación se configura al microcontrolador como dispositivo maestro y RTC como esclavo como se presenta en las siguientes figuras.

Figura 24. Dirección del RTC como esclavo en el bus I<sup>2</sup>C



Fuente: PALACIOS, Enrique, DOMINGUEZ, Fernando, LOPEZ, Lucas. Microcontrolador PIC16F84 Desarrollo de Proyectos.

Figura 25. Diagrama de conexión maestro (microcontrolador), esclavo (RTC)



Fuente: DS1307 Serial Real -Time Clock datasheet [Online]. Dallas Semiconductor.

El modo de manejo del bus I<sup>2</sup>C es de maestro en caso de que otros dispositivos de control, supervisión o almacenamiento se quieran agregar al proyecto, ya que es el microcontrolador el encargado de manejar toda la parte de comunicación respecto a sus periféricos esclavos.

La tabla siguiente, indica la ubicación de cada uno de los registros del reloj/calendario, y la función de cada uno de ellos con su respectivo rango de funcionamiento:

Tabla 9. Registros del reloj de tiempo real DS1307

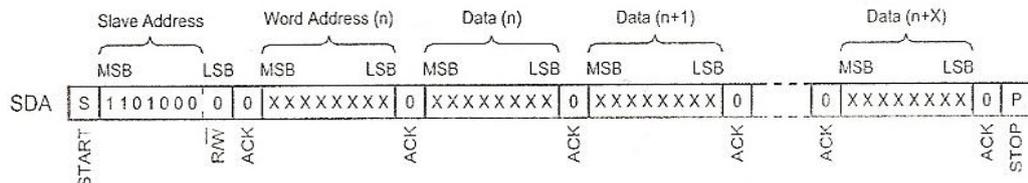
ADDRESS	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	FUNCTION	RANGE
00H	CH	10 Seconds			Seconds				Seconds	00-59
01H	0	10 Minutes			Minutes				Minutes	00-59
02H	0	12	10 Hour	10 Hour	Hours				Hours	1-12 +AM/PM 00-23
		24	PM/AM							
03H	0	0	0	0	0	DAY		Day	01-07	
04H	0	0	10 Date		Date			Date	01-31	
05H	0	0	0	10 Month	Month			Month	01-12	
06H	10 Year			Year				Year	00-99	
07H	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08H-3FH									RAM 56 x 8	00H-FFH

Fuente: DS1307 Serial Real -Time Clock datasheet [Online]. Dallas Semiconductor.

La escritura de datos en el RTC consiste en la actualización o iniciación de los registros de la hora y la fecha para que este comience a contar con su reloj independiente.

La transferencia de datos desde el microcontrolador al RTC sigue el procedimiento de escritura del maestro al esclavo del bus I<sup>2</sup>C, como se indica en la figura 26, para ello el microcontrolador envía la condición de inicio (bit start), luego envía la dirección del reloj (slave address) en modo de escritura que corresponde a 0xd0 en hexadecimal, después se envía la dirección del registro que se desea escribir (word address), para el RTC se iniciará la escritura desde la dirección 0x00 que corresponde al registro de segundos, esta dirección se incrementa automáticamente después de que se escribe cada uno de los registros, terminando con el registro de control 0x07. Al finalizar la escritura el microcontrolador envía la condición de paro.

Figura 26. Protocolo de escritura del maestro (microcontrolador) sobre el esclavo (RTC)



Fuente: PALACIOS, Enrique, DOMINGUEZ, Fernando, LOPEZ, Lucas. Microcontrolador PIC16F84 Desarrollo de Proyectos.

El registro de Control, dirección 07H, manipula la operación de la señal de salida de onda cuadrada, el bit 07 OUT controla el nivel lógico cuando SQWE = 0, el bit

04 SQWE habilita o deshabilita la salida de onda cuadrada, y RS1:RS0 controla la frecuencia de la onda cuadrada cuando SQWE = 1. La configuración dada para este registro se observa en la siguiente tabla.

Tabla 10. Configuración del registro de Control

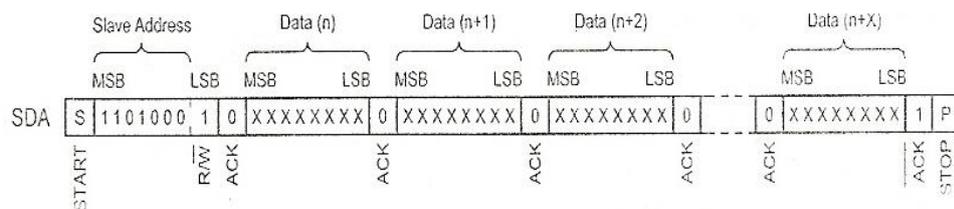
ADDRESS	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	FUNCTION
07H	OUT	0	0	SQWE	0	0	RS1	RS0	Control
07H	0	0	0	0	0	0	0	0	Control

En el registro de segundos, dirección 00H, el bit 07 CH es la habilitación del reloj, cuando este bit es fijado en 1 el oscilador es deshabilitado, cuando se escribe un 0 lógico el oscilador esta habilitado y pone en marcha el reloj.

En el registro de hora, dirección 02H, al momento de configurar la fecha, se debe especificar en que modo trabajar escribiendo en el bit 06, si escribe un 1 lógico, trabaja en el formato de 12 horas siendo el bit 05 el indicador de AM/PM, y los bits 04:00 la hora en formato BCD, si escribe un 0 lógico trabaja con el formato de 24 horas, siendo los bits 05:00 la hora en formato BCD.

La información de tiempo y calendario es obtenida al leer los registros apropiados, los cuales están localizados en las direcciones de memoria del RTC desde 0x00 hasta 0x07 como se muestra en la tabla 9. Esta lectura de datos del reloj por parte del microcontrolador sigue el procedimiento de lectura del esclavo por parte del maestro, como se indica en la figura 27.

Figura 27. Protocolo de lectura del maestro (microcontrolador) sobre el esclavo (RTC)



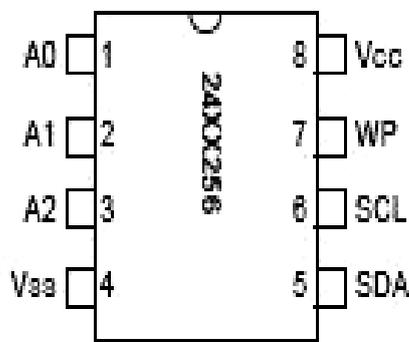
Fuente: PALACIOS, Enrique, DOMINGUEZ, Fernando, LOPEZ, Lucas. Microcontrolador PIC16F84 Desarrollo de Proyectos.

Para la lectura de datos el maestro envía el bit de inicio, luego transmite la dirección del RTC en modo de lectura que corresponde a 0xd1 y se comienza a leer los registros empezando en la dirección 0x00, esta se incrementa cada vez que se lee un registro del RTC, para finalizar la lectura se envía un bit de stop.

**2.2.4 Memoria externa serial EEPROM.** Los datos obtenidos por el receptor se almacenan en la memoria con referencia 24LC256<sup>20</sup> de MICROCHIP, esta tiene una capacidad de 256kbits (32kbytes), con la posibilidad de conectar hasta 8 dispositivos esclavos para conformar un banco de memoria de 2Mbits (256kbytes), el microcontrolador realiza la lectura o escritura de datos en la memoria utilizando la interfaz I<sup>2</sup>C, en cuyo caso a cada una de las memorias se le asigna una dirección diferente por hardware.

Al igual que el RTC la memoria EEPROM fue configurada como elemento esclavo para realizar la comunicación con el PIC

Figura 28. Encapsulado memoria EEPROM 24LS256



Fuente: EEPROM I2C 24LC256 datasheet [Online]. Microchip Technology Inc.

Los pines A0, A1 y A2 son terminales de entrada usadas para especificar la dirección del dispositivo esclavo en el bus I<sup>2</sup>C, debido a esto solo se admite hasta  $2^3 = 8$  memorias en el bus.

Dependiendo del estado lógico del pin WP (Protección de Escritura), se puede habilitar o deshabilitar la operación de escritura de la tabla de memoria, si se conecta a 0 lógico o se deja la terminal abierta, se puede realizar operaciones normales de escritura y lectura, si se fija a 1 lógico, únicamente realiza operaciones de lectura.

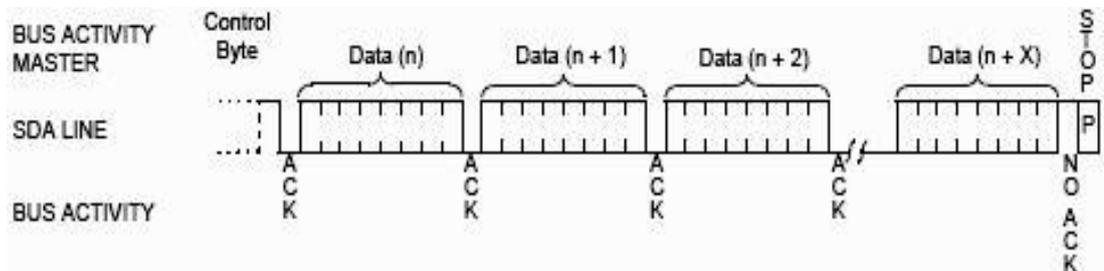
SCL y SDA: son los pines Serial Clock y Serial Data, del protocolo I<sup>2</sup>C.

Las especificaciones generales de la memoria se observan en la tabla 11.

<sup>20</sup>EEPROM I2C 24LC256 datasheet [Online]. Microchip Technology Inc. 2004. Disponible en Internet: <[http://www.microchip.com/stellent/idcplg?IdcService=SS\\_GET\\_PAGE&nodeId=2057&ty=&dt=Data+Sheets&section=Data+Sheets&ssUserText=24LC256](http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2057&ty=&dt=Data+Sheets&section=Data+Sheets&ssUserText=24LC256)>.



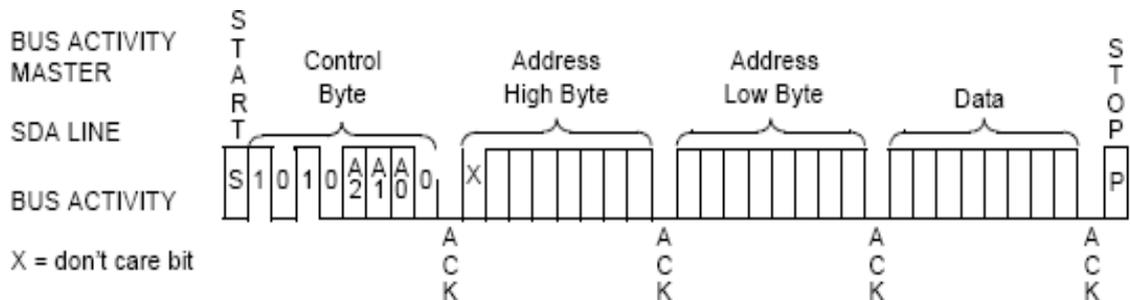
Figura 31. Lectura secuencial de datos



Fuente: EEPROM I2C 24LC256 datasheet [Online]. Microchip Technology Inc.

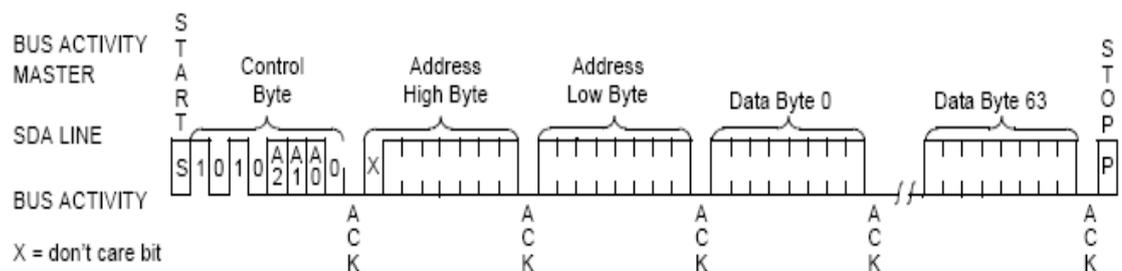
La escritura de datos en memoria abarca dos tipos de operaciones, escritura de un byte y escritura de una página de bytes, en el primero, el maestro ubica el dispositivo, acto seguido especifica la dirección a escribir y después el dato a escribir. En el segundo tipo de operación, el maestro en lugar de escribir un solo byte de dato, tiene la posibilidad de escribir hasta 64 bytes. El proceso de escritura de 1 byte tiene un retardo de 5ms según el fabricante.

Figura 32. Escritura de un byte



Fuente: EEPROM I2C 24LC256 datasheet [Online]. Microchip Technology Inc.

Figura 33. Escritura de varios bytes

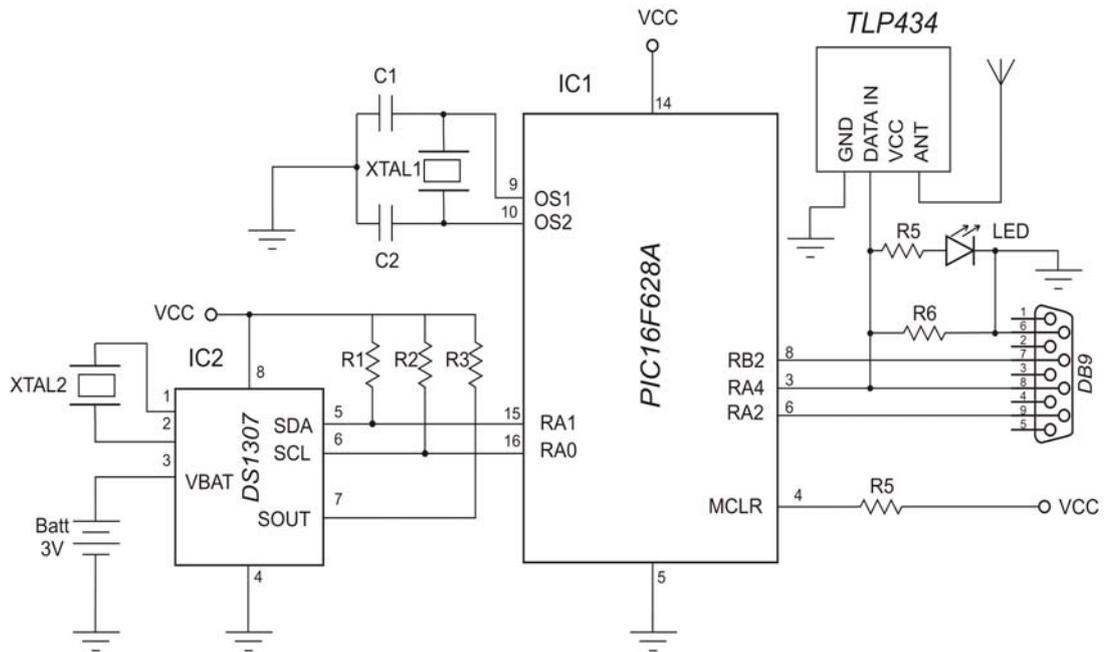


Fuente: EEPROM I2C 24LC256 datasheet [Online]. Microchip Technology Inc.

## 2.3 DESARROLLO DE SOFTWARE Y CONFIGURACION INTERNA DE LOS MICROCONTROLADORES

### 2.3.1 Programación y configuración módulo fijo.

Figura 34. Diagrama eléctrico módulo fijo

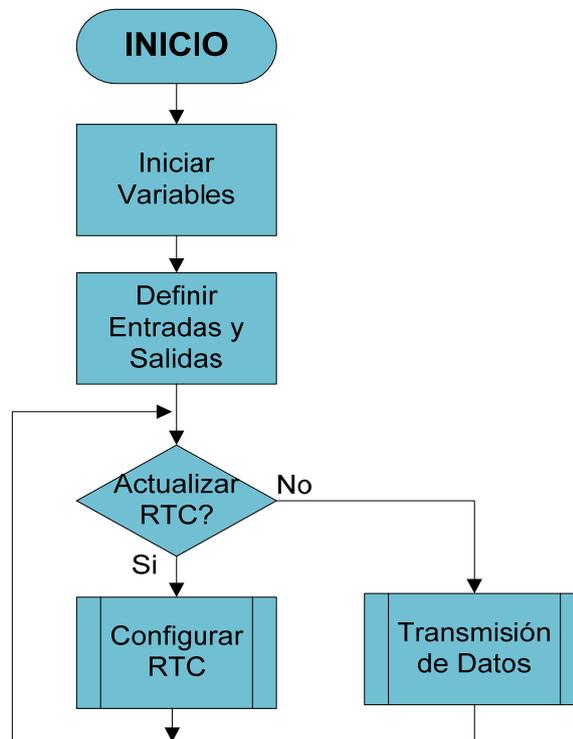


**2.3.1.1 Desarrollo de software implementado en microcontrolador.** La programación del microcontrolador se describe a través de diagramas de flujo básico, en forma resumida, más no representa una copia exacta de este.

- **Rutina Principal.** Como se menciona anteriormente el módulo fijo es el encargado de la lectura y escritura del RTC y además de gestionar la transmisión y recepción de datos (hora, minutos, segundos, día, mes, año, código de área), estas tareas se llevan a cabo gracias a la programación implementada en el microcontrolador. La programación de este se desarrollo en lenguaje C, por su versatilidad en el manejo de funciones y de los módulos del microcontrolador.

Como se observa en la figura 35, el programa comienza con el proceso de inicialización variables, además se define el nombre y dirección de los registros del RTC. La etapa de definición de entradas y salidas permite configurar los puertos del microcontrolador, según la tarea que vayan a realizar, estos están estipulados para el manejo de la comunicación con el módulo de transmisión inalámbrico, comunicación con el módulo fijo, lectura y escritura de RTC, control conexión entre otras.

Figura 35. Diagrama de flujo rutina principal microcontrolador Módulo Fijo.

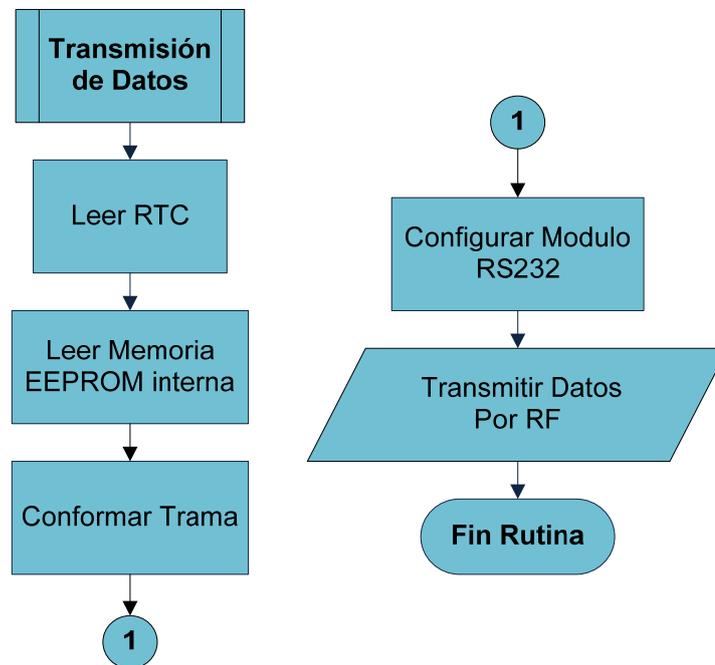


Cuando el módulo fijo no se encuentra conectado al dispositivo de configuración, entra a la subrutina de transmisión de datos, en esta se lee el RTC y se conforma la trama de datos que se envía en forma serial al módulo transmisor de RF. Sí se conecta al módulo de configuración, se cambia a nivel alto un pin del microcontrolador configurado como entrada, lo que permite al PIC ingresar a la subrutina de configuración del reloj.

• **Subrutina Transmisión de Datos.** En esta subrutina se conforma y se transmite la trama vía RF al módulo móvil. Para acceder a los datos del RTC primero se establece la comunicación maestro esclavo entre el microcontrolador y el reloj calendario, se configura el RTC en modo de lectura y se procede a leer secuencialmente los primeros registros de datos del RTC (0x0H) hasta RTC (0x6H), estos corresponden a:

- Segundos = RTC (0x0H)
- Minutos = RTC (0x1H)
- Hora = RTC (0x2H)
- Día = RTC (0x3H)
- Fecha = RTC (0x4H)
- Mes = RTC (0x5H)
- Año = RTC (0x6H)

Figura 36. Diagrama de flujo subrutina transmisión de datos



El código de área se lee de la memoria EEPROM del microcontrolador, este se encuentra almacenado en los registros 0x03, 0x04 y 0x05 como se muestra en la siguiente figura.

Figura 37. Mapa de memoria EEPROM PIC16F628A

PIC CPU EPROM Memory - PIC16F628A MOD FIJO												
00	FF	FF	FF	00	00	FF						
10	FF											
20	FF											
30	FF											
40	FF											
50	FF											
60	FF											
70	FF											

Cuando se termina el proceso de lectura de datos se conforma la trama que tiene una longitud de 13 bytes fijos como se muestra a continuación.

***h1, h1, h2, horas, minutos, segundos, error, cod, fecha, mes, año, error1, h2***

*h1*: este byte permiten establecer la comunicación entre los módulos de radiofrecuencia trasmisor receptor.

*h2*: indica el inicio y el final de la trama de datos.

*Error*: conformado por la suma de los bytes de horas, minutos y segundos.

*Error1*: conformado por la suma de los bytes de cod, fecha, mes, año.

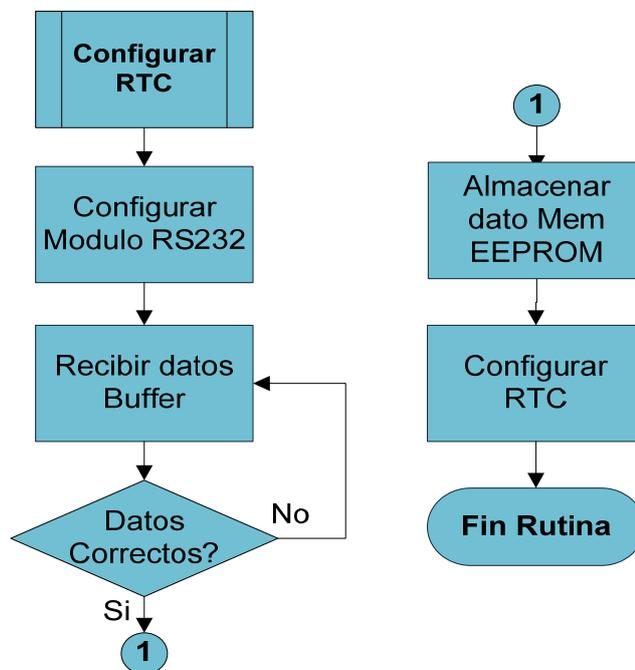
Los bytes h2, error, error1 se envían para realizar corrección de errores en el módulo móvil.

La comunicación entre el módulo transmisor inalámbrico y el microcontrolador se realiza por el puerto serial USART, que se configura en modo de transmisión asíncrona con una velocidad de transmisión de 2.4kbps.

• **Subrutina configuración de RTC.** Al conectar el dispositivo de configuración, el microcontrolador entra en la subrutina de configuración de RTC, la comunicación entre los dos módulos es física y usa interfaz de comunicación RS232.

El dispositivo de configuración envía datos con una longitud de trama fija de 13 bytes con las mismas características de la trama que envía el módulo fijo. Como se menciona anteriormente, para la recepción de los datos se configura el módulo de transmisión serial UART. En el microcontrolador se asigna un espacio de memoria formado por un buffer o vector en el cual se almacena la trama de datos recibida.

Figura 38. Diagrama de flujo subrutina configuración RTC



La corrección de errores de la trama recibida se realiza de la siguiente manera: una vez almacenados los datos se procede a verificar los bytes h2 enviados por el módulo fijo, si estos son correctos se procede a sumar horas, minutos, segundos, el resultado se almacena en una variable temporal error2, se suma también el código, fecha, mes, año y se almacena en la variable error3, estos valores se los compara con los bytes error y error1 de la trama recibida, si estos no son iguales se borra el buffer y se espera nuevamente la trama.

Si los datos almacenados en el buffer son correctos, el microcontrolador procede a almacenar el cod o código de área (lugar de ubicación del módulo) en la memoria EEPROM. Y los bytes de fecha, mes, año, horas, minutos, segundos se escriben en los registros del RTC configurado en modo de escritura.

**2.3.1.2 Configuración de registros del microcontrolador.** A continuación se describe los registros y parámetros configurados en el microcontrolador PIC16F628A según los módulos utilizados.

- **Configuración del oscilador.**

Tabla 12. Configuración del registro CONFIG

$\overline{CP}$	—	—	—	—	$\overline{CPD}$	LVP	BOREN	MCLRRE	FOSC2	$\overline{PWRTE}$	WDTE	FOSC1	FOSC0
0					0	0	0	1	0	0	0	0	1
bit 13													bit 0

En este registro se determina la frecuencia del bus de datos y señales de reloj, el microcontrolador puede operar de 8 maneras diferentes de oscilación, los cuales son seleccionados con tres bits de configuración (FOSC2, FOSC1, FOSC0), del registro CONFIG estos modos son:

- ✓ LP Cristal de baja frecuencia.
- ✓ XT Cristal o cerámico resonador.
- ✓ HS Cristal o cerámico resonador de alta velocidad.
- ✓ RC Resistencia/condensador externos (2 modos).
- ✓ INTOSC Oscilador de precisión interno (2 modos).
- ✓ EC entrada de reloj externa.

El modo seleccionado es XT cristal o cerámico resonador. El cristal seleccionado es de 4Mhz, según el fabricante los valores de los condensadores que acompañan al cristal deben ser de 15 a 30pF. Una capacitancia grande produce mayor estabilidad del cristal, pero hace que se incremente el tiempo de arranque. La selección de los condensadores es hecha de acuerdo a la tabla 13, al ser la frecuencia de 4Mhz nos permite elegir los C1 = C2 = 22pF:

Tabla 13. Rango de valores de frecuencia y capacitancia

Mode	Freq	OSC1(C1)	OSC2(C2)
LP	32 kHz	15-30 pF	15-30 pF
	200 kHz	0-15 pF	0-15 pF
XT	100 kHz	68-150 pF	150-200 pF
	2 MHz	15-30 pF	15-30 pF
	4 MHz	15-30 pF	15-30 pF
HS	8 MHz	15-30 pF	15-30 pF
	10 MHz	15-30 pF	15-30 pF
	20 MHz	15-30 pF	15-30 pF

Fuente: PIC16F628A datasheet [Online]. Microchip Technology Inc.

• **Configuración del módulo USART modo asíncrono.** La comunicación entre el módulo fijo y módulo de configuración y, la transmisión de datos entre el módulo fijo y el transmisor de RF se realiza por comunicación serial RS-232. El microcontrolador posee un módulo de USART, que hace posible esta comunicación.

Tabla 14. Configuración del registro TXSTA

CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D
0	0	0	0		0	0	0

bit 7 bit 0

TX9 = 0 permite la selección de una trama fija de 8 bits para la transmisión de datos por el puerto RS232, SYNC = 0 configura un modo asíncrono de transmisión.

Tabla 15. Configuración del registro RCSTA

SPEN	RX9	SREN	CREN	ADEN	FERR	OERR	RX9D
1	0	0	0	0	0	0	0

bit 7 bit 0

SPEN=1 habilita el puerto serial, es decir configura los pines RX/DT y TX/CK del microcontrolador como pines de recepción y transmisión. RX9=0 configura una trama fija de 8 bits para la recepción de datos.

Tabla 16. Configuración del registro SPBRG

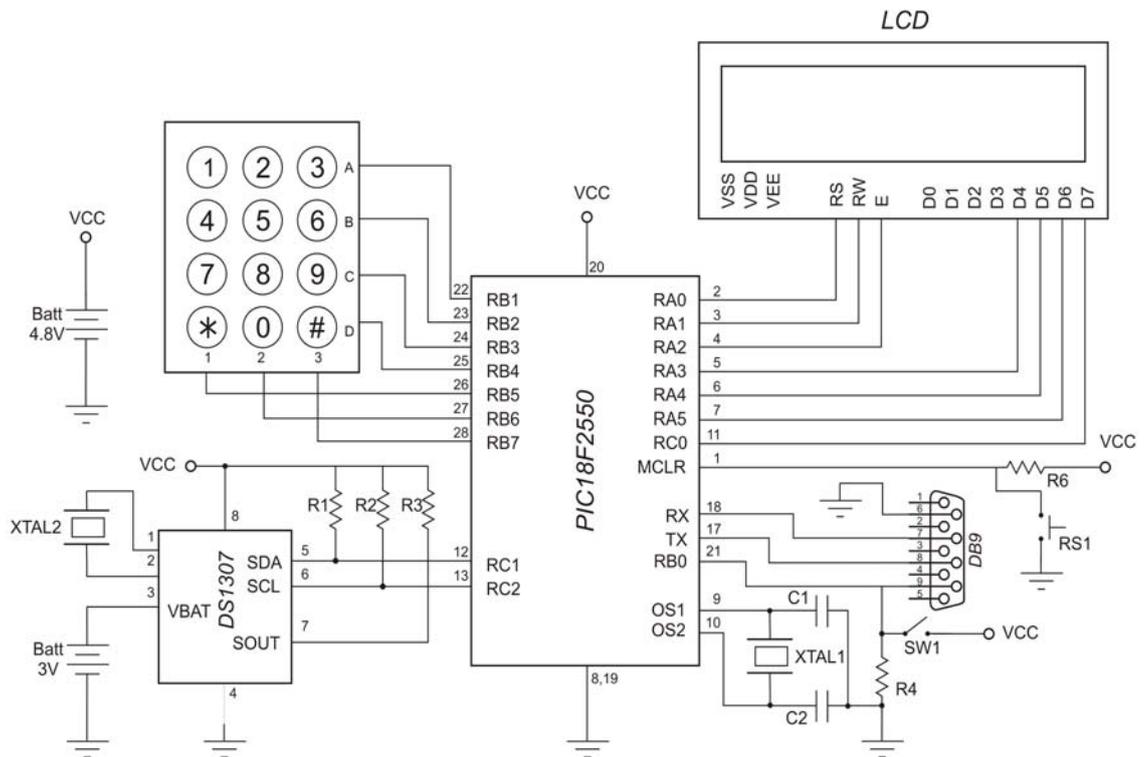
0	0	0	1	1	0	0	1
---	---	---	---	---	---	---	---

bit 7 bit 0

Los valores binarios de este registro permiten asignarle un valor decimal al SPBRG de 25, con este valor se obtiene una velocidad de transmisión de 2.4Kbps. Después de realizar varias pruebas de comunicación se encuentra que a una velocidad de transmisión de 2.4Kbps, el error en la comunicación entre transmisor y receptor de RF es menor.

### 2.3.2 Programación y configuración módulo de configuración.

Figura 39. Diagrama eléctrico módulo de configuración

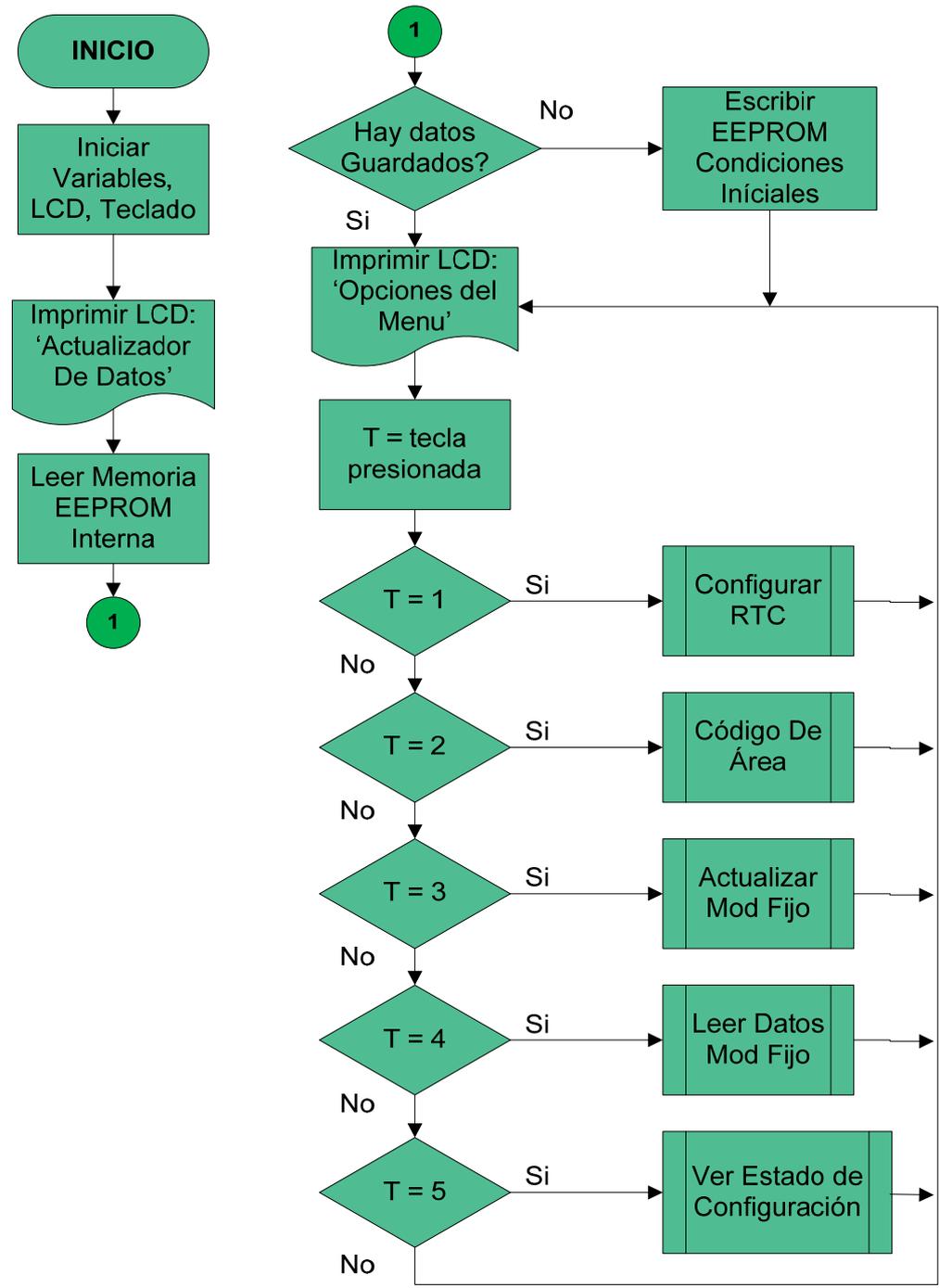


**2.3.2.1 Desarrollo de software implementado en microcontrolador.** Como se menciona anteriormente el microcontrolador PIC18F2550 se encarga de gestionar la programación del RTC y de la actualización del módulo fijo.

- **Rutina Principal.** Al comenzar el programa realiza el proceso de inicializar variables, pantalla LCD y teclado, en donde se fijan las variables en condición "0", declaradas para el funcionamiento del módulo, configura los pines del microcontrolador para usar la pantalla LCD y el teclado matricial. enseguida el programa despliega en la pantalla LCD un menú de opciones en el que se puede seleccionar con el teclado matricial, entre las que están: configuración de hora,

fecha y código, la transmisión y recepción de datos del módulo fijo. Cada tarea del menú se realiza en una subrutina diferente.

Figura 40. Diagrama de flujo rutina principal microcontrolador módulo de configuración.



- **Subrutina Configurar RTC.** Al momento que se presiona la tecla 1, el programa ejecuta la subrutina de configuración de registros del reloj de tiempo real. En esta rutina se implementa un proceso de seguridad para la programación de la hora y la fecha, se requiere el ingreso de una contraseña de 4 dígitos, muestra en la pantalla LCD un submenú con dos opciones que son cambio de clave y modificación de registro del RTC.

En la opción 1, se permite el cambio de la clave anterior almacenada en la memoria EEPROM (la contraseña esta guardada desde la dirección 0x02 hasta la 0x05 ver figura 41), el microcontrolador despliega en la pantalla LCD el mensaje 'ingrese clave anterior', al introducir la clave el microcontrolador la compara con la almacenada en su memoria, sí no son iguales el programa finaliza la subrutina y retorna a la rutina principal. Sí la contraseña anterior es correcta se ingresa a través del teclado dos veces la nueva contraseña, si son idénticas se la almacena en la memoria EEPROM y finaliza la subrutina.

Figura 41. Mapa de memoria EEPROM PIC18F2550

PIC18 CPU EPROM Memory - PIC18F2550 MOD CONFIGURACION															
00	FF	FF	01	02	03	04	FF	00	00	00	FF	FF	FF	FF	FF
10	FF														
20	FF														
30	FF														
40	FF														
50	FF														
60	FF														
70	FF														
80	FF														
90	FF														
A0	FF														
B0	FF														
C0	FF														
D0	FF														
E0	FF														
F0	FF														

En la opción 2 del menú clave, se puede configurar los registros del reloj, solicita el ingreso de la clave de acceso, si esta es correcta el programa hace el registro secuencialmente de horas, minutos, segundos, fecha, mes y año. Cada dato se encuentra validado para que se ingresen valores correctos. Por ejemplo, en el caso del dato de hora se debe ingresar entre 00 y 23, si se introduce valores fuera de este rango el programa vuelve a solicitar este dato, al ingresar el valor correcto se lo almacena en una variable temporal, y se pide ingresar el dato siguiente finalizando con el ingreso del año. Una vez se introduzcan todos los datos, se muestra en la pantalla LCD la configuración realizada y se escriben los registros del RTC.

Las figuras 42 y 43 muestran el diagrama de flujo del la subrutina de configuración de reloj de tiempo real.

Figura 42. Diagrama de flujo subrutina configurar RTC (a)

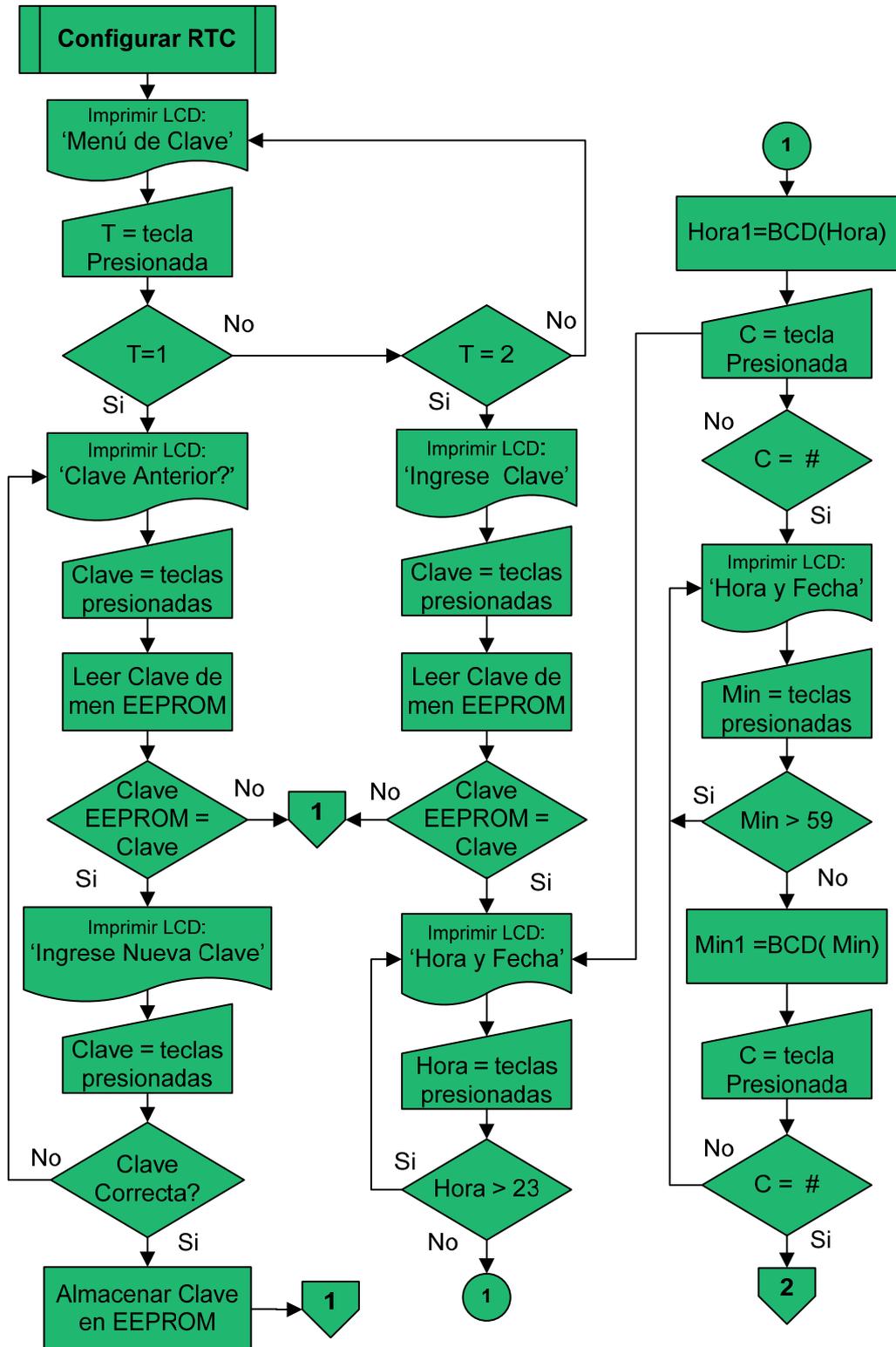
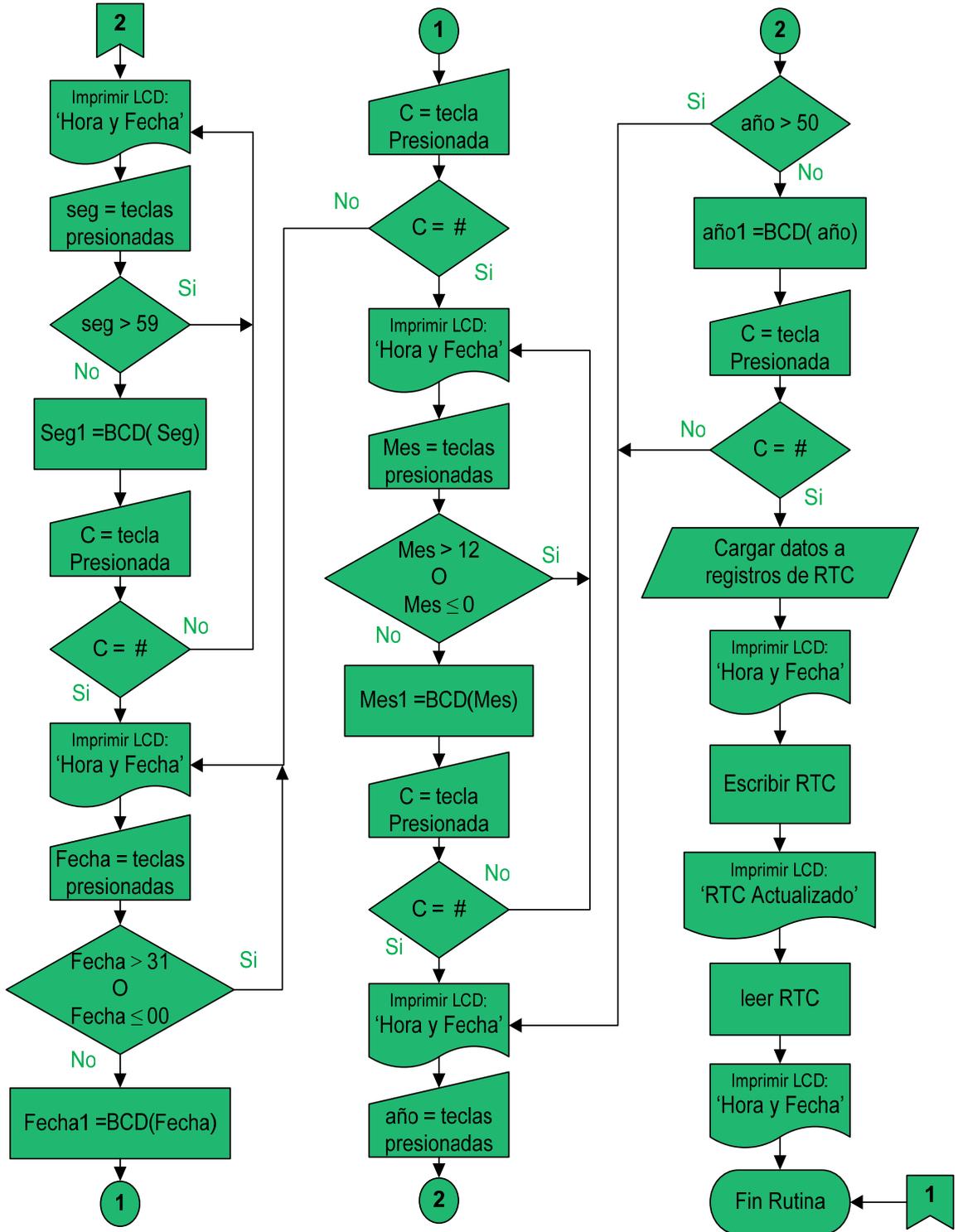
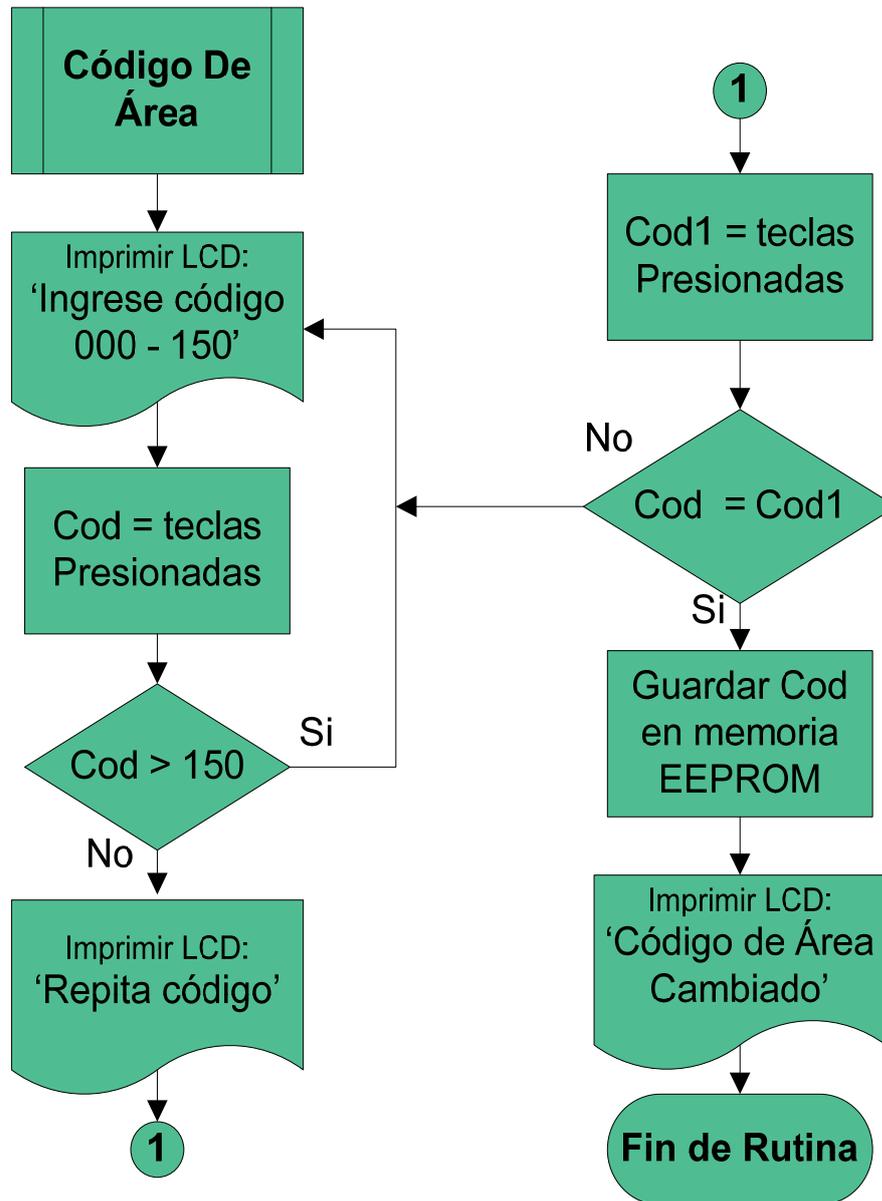


Figura 43. Diagrama de flujo subrutina configurar RTC (b)



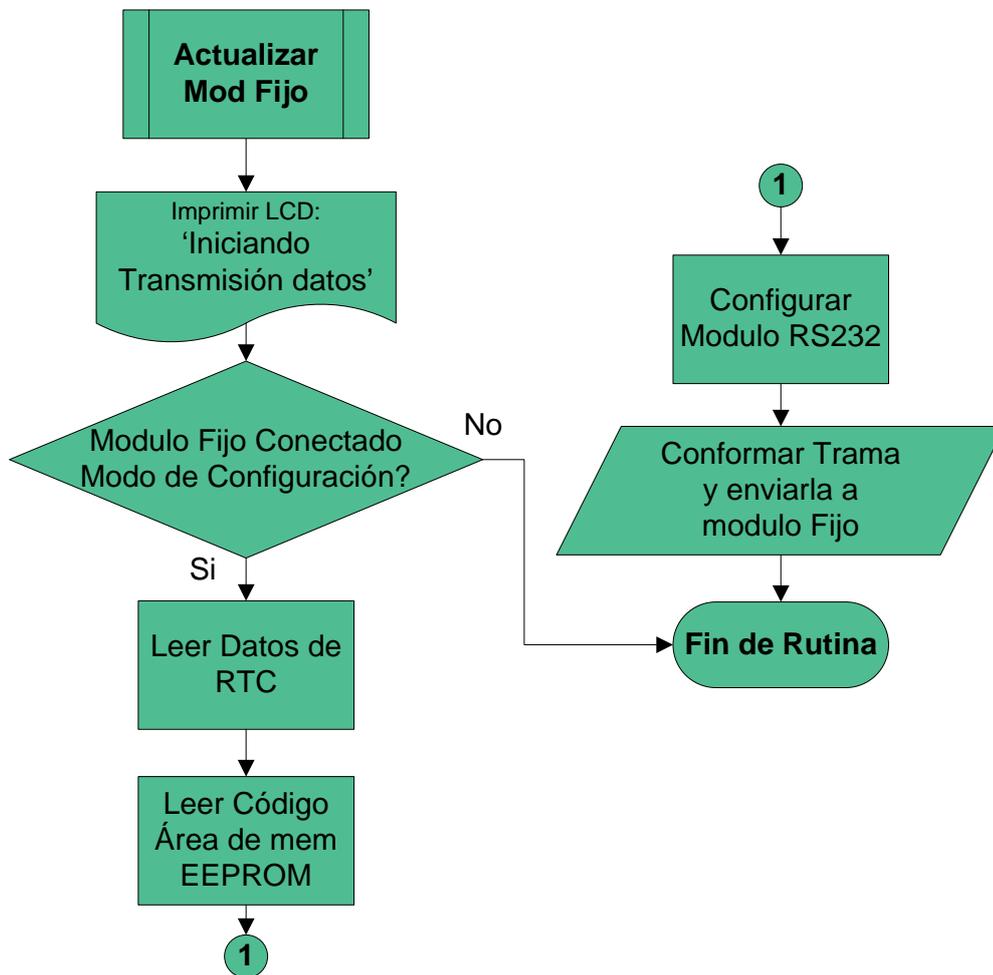
- **Subrutina código de área.**

Figura 44. Diagrama de flujo subrutina cambio de código de área



La segunda opción del menú principal, permite modificar el código de área que será transmitido al módulo fijo, el código se ingresa dos veces, si es introducido correctamente se almacena en la memoria EEPROM (el código se encuentra almacenado desde la dirección de memoria 0x07 hasta 0x08, ver figura 41), de lo contrario se solicita ingresarlo nuevamente.

Figura 45. Diagrama de flujo subrutina actualizar módulo fijo



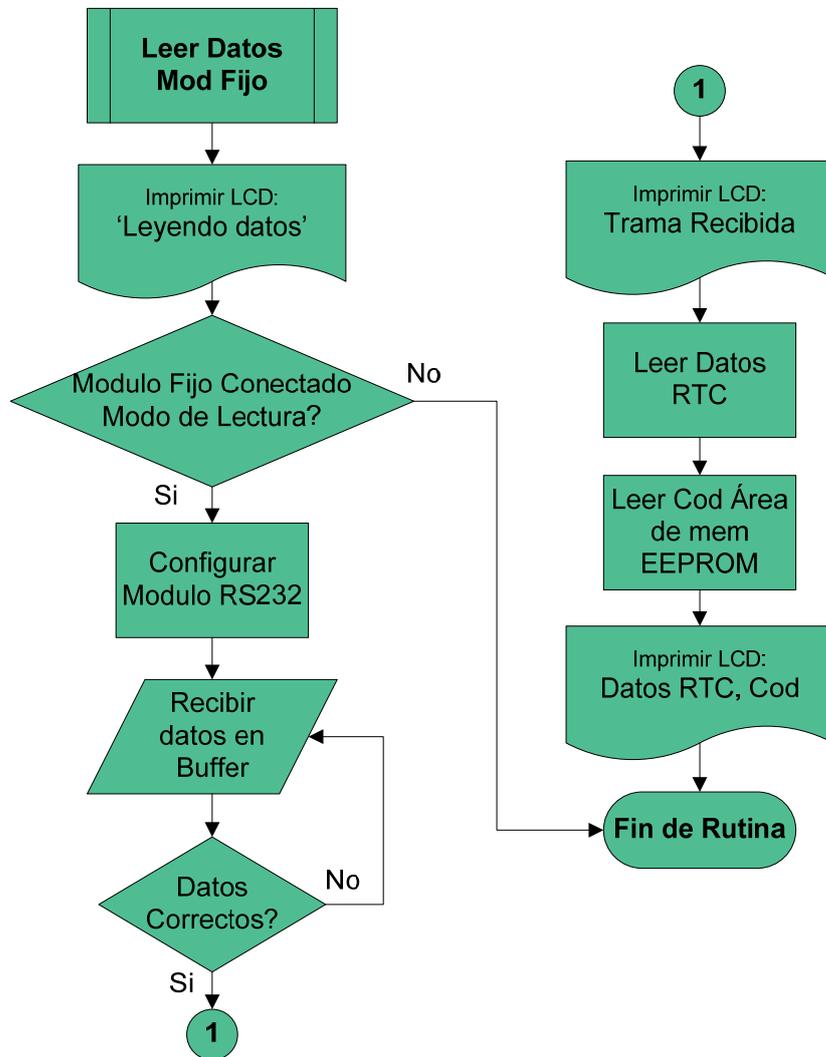
La tercera opción del menú principal permite la transmisión de datos para la configuración del RTC del módulo fijo.

Para transmitir los datos, el microcontrolador verifica si el módulo fijo se encuentra conectado al módulo de configuración en modo de escritura, sí es así lee los registros del RTC y el código de área almacenado en la memoria EEPROM y se conforma la trama que será enviada usando protocolo RS-232 a través del módulo de transmisión serial UART. La trama enviada tiene las mismas características que la transmitida por el módulo fijo.

***h1, h1, h2, horas, minutos, segundos, error, cod, fecha, mes, año, error1, h2***

La explicación de cada uno de los bytes de la trama se realiza en la sección 2.3.1.1

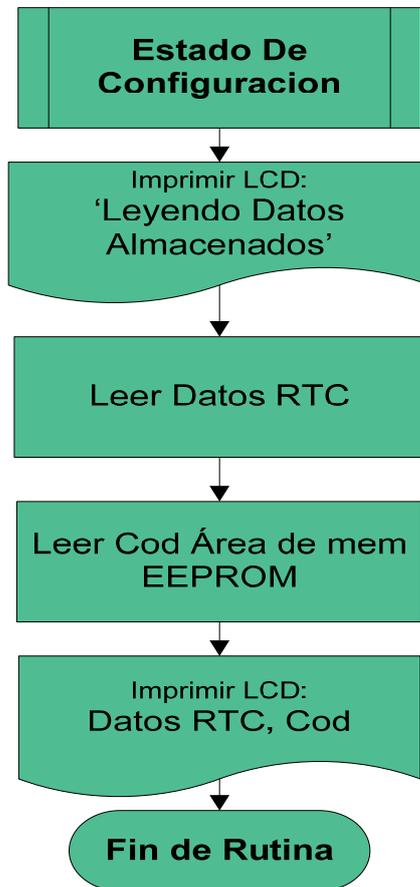
Figura 46. Diagrama de flujo subrutina leer datos módulo fijo



La cuarta opción del menú principal permite la lectura de datos que transmite el módulo fijo.

Para leer los datos el microcontrolador verifica, si el módulo fijo se encuentra conectado al módulo de configuración en modo de lectura, se asigna un espacio de memoria formado por un buffer en el que se almacena la trama de datos recibida. Se realiza corrección de errores a la trama, si los datos son correctos se muestran en la pantalla LCD, luego se leen los datos del RTC, el código de área del módulo de configuración y se imprimen en pantalla para verificar si están sincronizados con los datos del módulo fijo.

Figura 47. Diagrama de flujo subrutina estado de configuración



La última opción del menú principal permite conocer el estado de configuración del módulo, permitiendo mostrar en la pantalla LCD la fecha y hora del RTC y el código de área almacenado en la memoria EEPROM.

**2.3.2.2 Configuración de registros del microcontrolador.** A continuación se describe la configuración de los registros del microcontrolador PIC18F2550, utilizados para el funcionamiento de los módulos: oscilador, USART e I<sup>2</sup>C.

- **Configuración del oscilador.** El oscilador del PIC18F2550 incorpora múltiples fuentes de reloj que se pueden dividir en tres tipos; cristales externos, fuentes externas del reloj conectadas al timer1 y el bloque del oscilador interno. Estas tres fuentes hacen que el oscilador disponga de doce modos distintos de funcionamiento, los cuales se seleccionan a través de dos registros de configuración (CONFIG1L y CONFIG1H).

XT            Cristal/resonador cerámico.

XTPLL	Cristal/resonador cerámico con PLL activo.
HS	Cristal de alta velocidad/resonador cerámico.
HSPLL	Cristal de alta velocidad/resonador cerámico con PLL activo
EC	Reloj externo con salida fosc/4.
ECIO	Reloj externo con I/O en RA6.
ECPLL	Reloj externo con PLL activo y salida fosc/4 en RA6.
ECPIO	Reloj externo con PLL activo, I/O en RA6.
INTHS	Oscilador interno usado como fuente de reloj del microcontrolador, Oscilador HS usado como fuente de reloj del USB.
INTXT	Oscilador interno usado como fuente de reloj del microcontrolador, Oscilador XT usado como fuente de reloj del USB.
INTIO	Oscilador interno usado como fuente de reloj del microcontrolador, Oscilador EC usado como fuente de reloj del USB, I/O digital en RA6.
INTCKO	Oscilador interno usado como fuente de reloj del microcontrolador, Oscilador EC usado como fuente de reloj del USB, salida fosc/4 en RA6.

De los modos de funcionamiento existentes, se escoge el oscilador primario XT, la frecuencia es de 4 Mhz dada por un cristal resonador, conectado como se muestra en la figura siguiente, los valores de los condensadores son los recomendados por el fabricante  $C = 27\text{pF}$ .

En el anexo H, se muestra el esquema del módulo oscilador, indicando la fuente utilizada (oscilador primario) y los valores que toman los distintos bits de estos registros.

Tabla 17. Configuración del registro CONFIG1H

IESO	FCMEN	—	—	FOSC3	FOSC2	FOSC1	FOSC0
0	0			0	0	0	0
bit 7							bit 0

El bit IESO permite el intercambio de oscilador interno a externo, por lo que se deshabilitará esta opción colocando el bit en 0. Los bits FOSC0, FOSC1, FOSC2 y FOSC3 seleccionan el tipo de oscilador (primario, secundario o interno), para escoger el modo primario XT se configuran estos bits en 0.

Tabla 18. Configuración del registro CONFIG1L

—	—	USBDIV	CPUDIV1	CPUDIV0	PLLDIV2	PLLDIV1	PLLDIV0
		1	0	0	0	0	0
bit 7							bit 0

Al seleccionar el oscilador primario XT, los CPUDIV0 Y CPUDIV1, permiten elegir la post-escala para el reloj del sistema. Al configurar los bits CPUDIV0 = 0 Y CPUDIV1 = 0 el oscilador primario es usado directamente. La señal del reloj no es dividida.

- **Configuración del módulo EUSART modo asíncrono.** La transmisión o recepción de datos entre el módulo fijo y el módulo de configuración se hace por comunicación serial RS-232. El microcontrolador posee un módulo de EUSART que es controlado al configurar los registros TXSTA, RCSTA y BAUDCON.

Los pines del módulo enhanced USART están configurados en el puerto c del microcontrolador en el orden siguiente: RC6/TX/CK (transmisión) y RC7/RX/DT/SDO (recepción).

Tabla 19. Configuración del registro TXSTA

CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D
0	0	0	0	0	0	0	0

bit 7 bit 0

TX9 = 0 permite la selección de una trama fija de 8 bits para la transmisión de datos por el puerto RS232, SYNC = 0 configura un modo asíncrono de transmisión.

Tabla 20. Configuración del registro RCSTA

SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
1	0	0	0	0	0	0	0

bit 7 bit 0

SPEN = 1 habilita el puerto serial, es decir configura los pines RX/DT y TX/CK del microcontrolador como pines para el puerto serial. RX9 = 0 configura una trama fija de 8 bits para la recepción de datos.

Tabla 21. Configuración del registro BAUDCON

ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN
0	0		0	0		0	0

bit 7 bit 0

El bit BRG16 permite configurar los registros de generación de baudios: SPBRG (8 o 16bits), SPBRGH (16 bits)

El BRG16 = 0 configura los registros generadores de baudios de 8 bits, solo tiene en cuenta el valor del registro SPBRG, mientras que el SPBRGH no interesa.

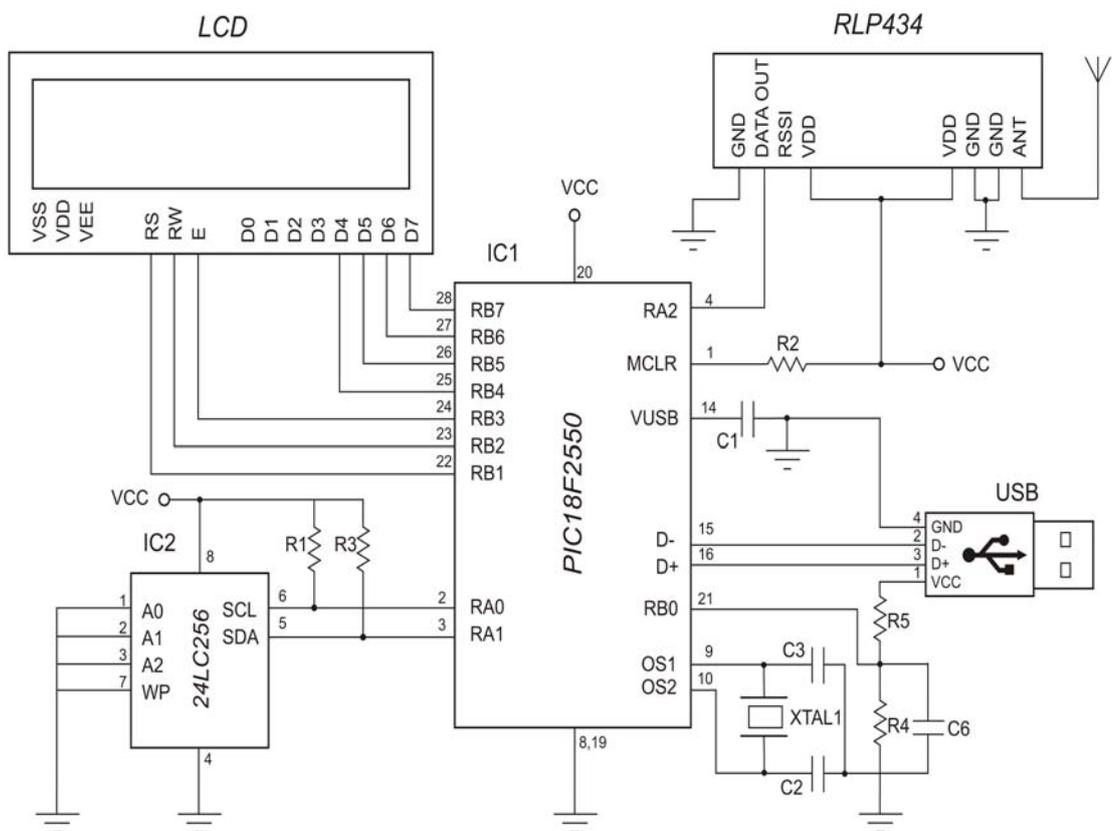
Tabla 22. Configuración del registro SPBRG

0	0	0	1	1	0	0	1
bit 7							bit 0

Para generar una velocidad de transmisión de 2.4Kbps se debe asignar al registro SPBRG el valor 25 en decimal o 11001 en binario.

### 2.3.3 Programación y configuración módulo móvil.

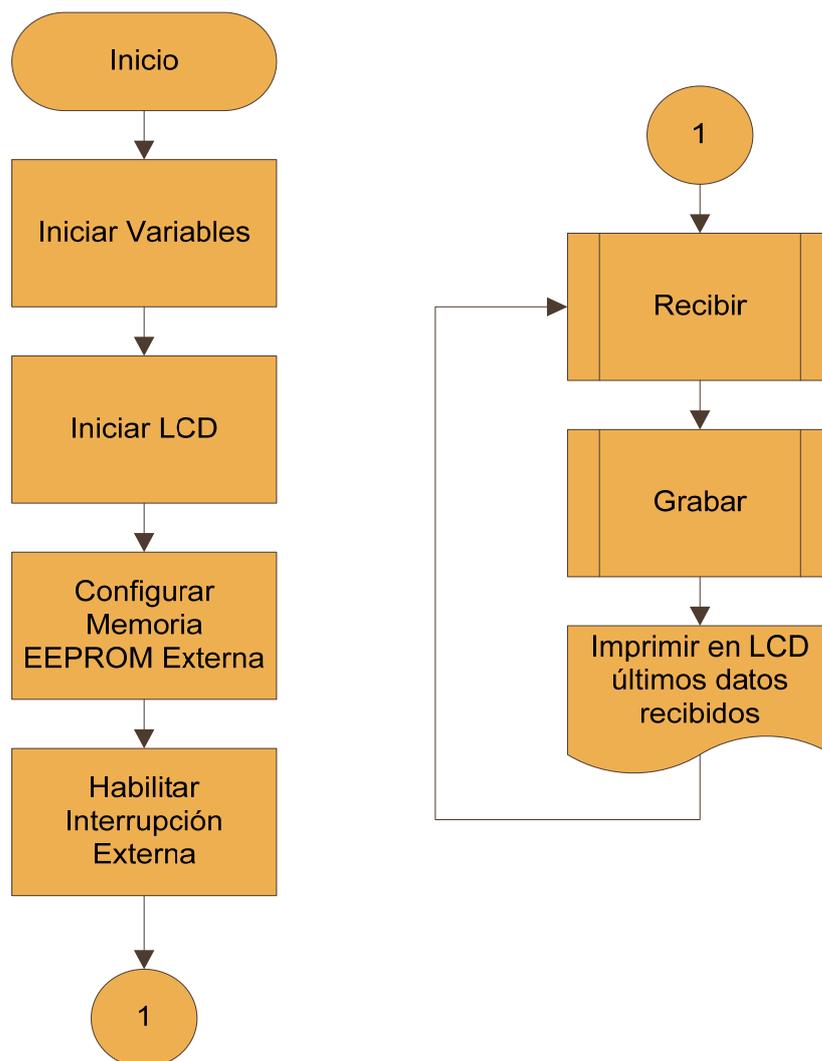
Figura 48. Diagrama eléctrico módulo móvil



### 2.3.3.1 Desarrollo de software implementado en microcontrolador.

- **Rutina Principal.** En el programa principal se configura los registros especiales utilizados, sus puertos, se establece las variables y subrutinas necesarias para su funcionamiento.

Figura 49. Diagrama de flujo rutina principal microcontrolador Módulo Móvil.



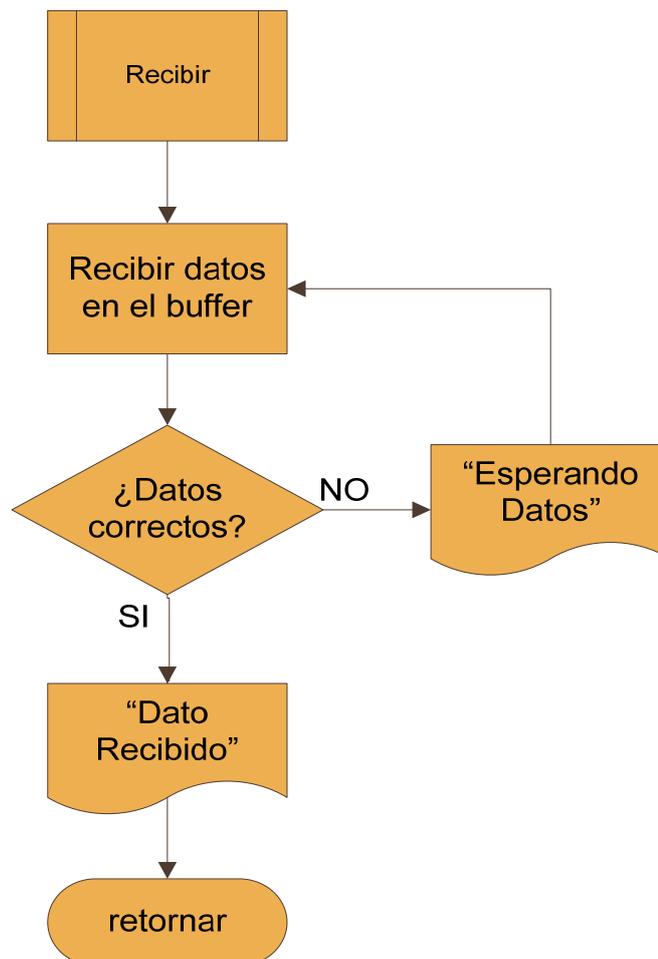
- **Subrutina Recibir.** Una vez que el programa realiza las configuraciones necesarias pasa directamente a la subrutina recibir.

La subrutina *recibir* cuenta con un buffer *DATOS* en el cual recibe todo dato para luego ser analizado, si el dato es incorrecto imprime en el LCD el mensaje

“Esperando Datos”, sí el dato es correcto sale de la subrutina *recibir* y luego el programa principal se dirige a la subrutina *grabar*.

Para saber sí los datos recibidos son correctos se realiza la verificación en dos pasos, teniendo en cuenta que la comunicación de los dos módulos (fijo y móvil) es de tipo Half Duplex.

Figura 50. Diagrama de flujo de la subrutina recibir



Primero se compara el inicio y el final de la trama para establecer si son datos posiblemente válidos o es simplemente ruido:

```
if(DATOS[3]=='F' && DATOS[4]=='E' && DATOS[5]=='B' && DATOS[27]=='E' && DATOS[28]=='B')
```

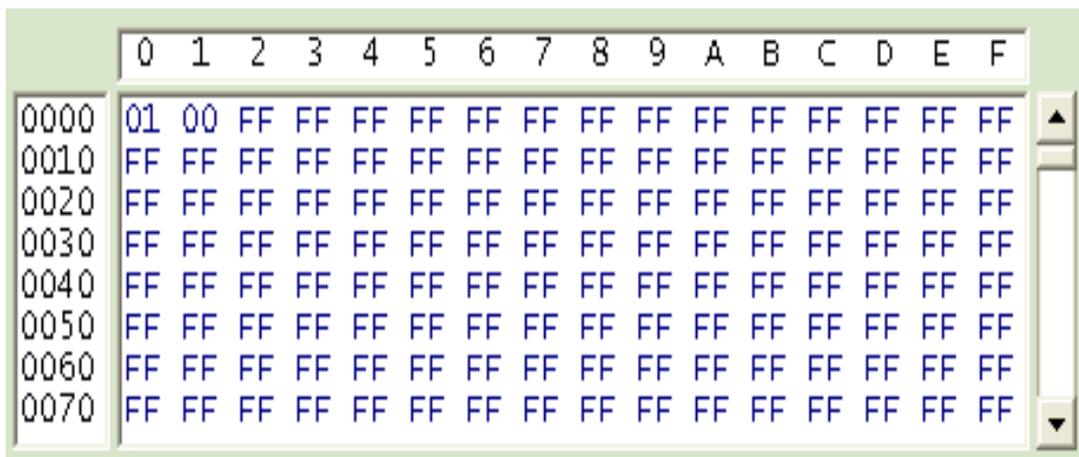
En segundo lugar, una vez superada la primera validación, se procede a convertir la cadena de caracteres a su valor numérico y se suma tanto la fecha (día, mes,

año) con el código y la hora (horas, minutos, segundos), los resultados de las operaciones se comparan con los valores que se obtuvieron antes de enviarlos y que están contenidos en la trama. Sí los resultados coinciden se puede afirmar que el dato recibido es correcto y se puede continuar con el proceso.

- **Subrutina Grabar.** se procede a grabar los datos en la memoria EEPROM externa organizada así: los dos primeros registros conforman un puntero de 16 bit para el manejo de la dirección de memoria que ha utilizado y del registro 3 al 32768 para almacenar los datos recibidos.

Cuando el dispositivo se pone en marcha por primera vez su memoria se encuentra vacía, es decir todos los bits de los registros se encuentran en nivel alto (valor FF en hexadecimal), es necesario inicializar el puntero por medio de software, como se muestra en la siguiente figura.

Figura 51. Parte inicial del mapa de memoria de externa 24LC256

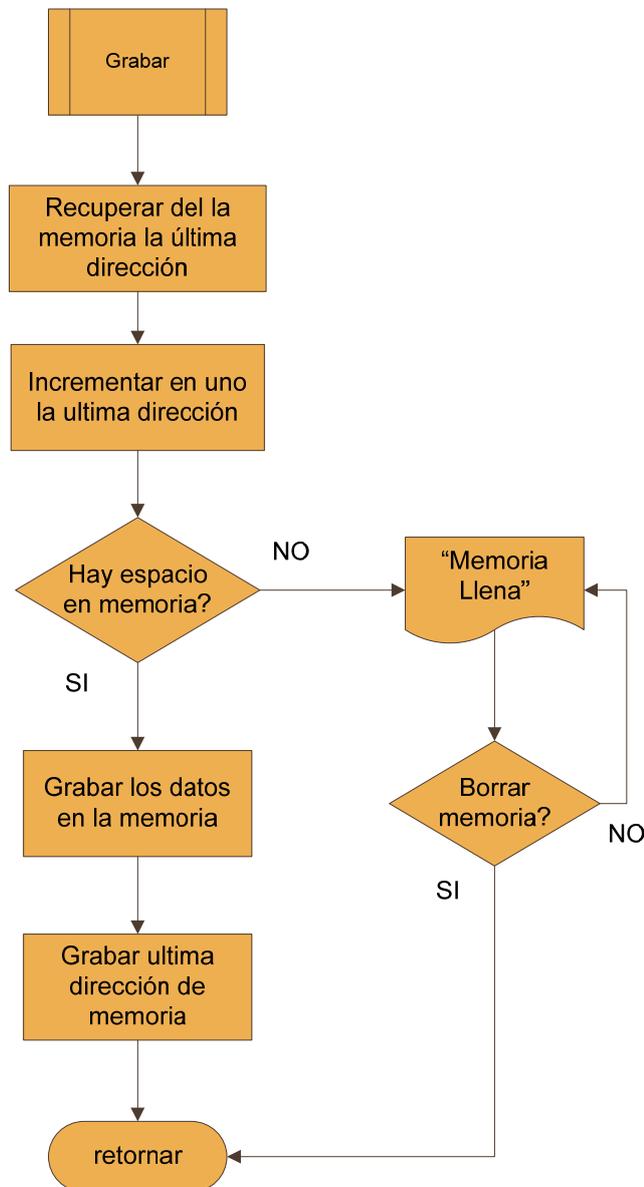


	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	01	00	FF													
0010	FF															
0020	FF															
0030	FF															
0040	FF															
0050	FF															
0060	FF															
0070	FF															

Al inicializar los punteros el dispositivo móvil queda listo para almacenar los datos, para ello se cuenta con la subrutina *grabar*, en el primer paso recupera la dirección de memoria que se utiliza en algún proceso anterior y lo incrementa en uno para escribir el dato nuevo, antes de escribir, verifica si hay espacio en memoria comparando el valor del puntero con el total de la capacidad de memoria, sí lo hay procede a grabar todos los datos y al final guarda la última dirección de memoria que usa y vuelve al programa principal; sí no hay espacio en memoria imprime en el LCD un mensaje de “*Memoria Llena*” y, solo continuará el proceso cuando ésta se borre nuevamente.

Para escribir en la memoria externa se utiliza el comando *write\_ext\_eeprom (dir,mem )*, donde *dir* es la dirección de memoria en la que se guarda y *mem* es el dato que se va a grabar.

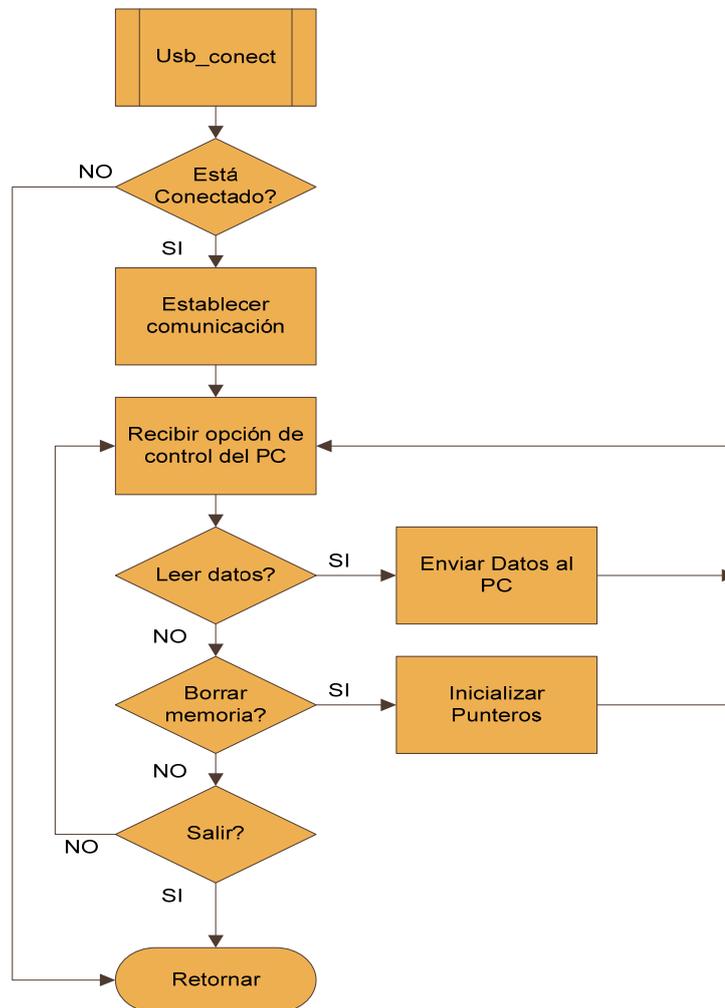
Figura 52. Diagrama de flujo de la subrutina grabar



Adicionalmente, se implementa una subrutina de interrupción para manejo de la interfaz USB con el computador, tiene como fin la recuperación y visualización de los datos almacenados en la memoria externa 24LS256, como se menciona anteriormente para inicializar el puntero (borrar memoria).

- **Subrutina usb\_conect.** El módulo móvil entra a la subrutina de interrupción, únicamente cuando esté conectado físicamente al computador, porque se configura en el microcontrolador una interrupción externa.

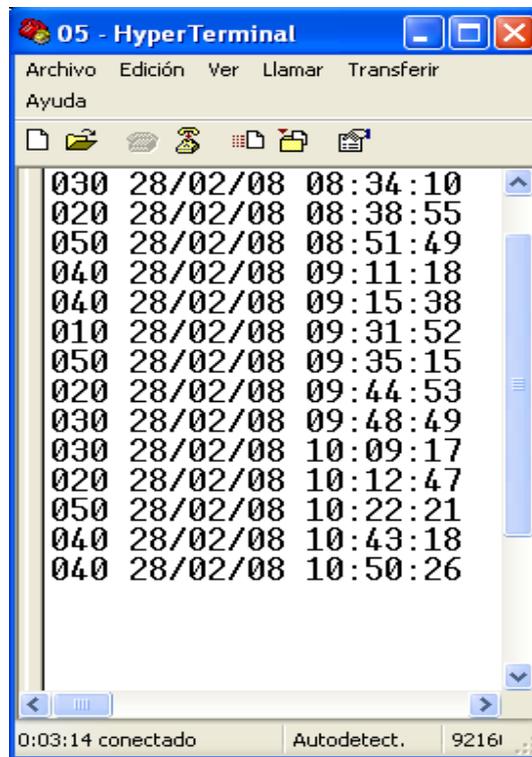
Figura 53. Diagrama de flujo de la interrupción usb\_conect



La interrupción se ejecuta por cambio de nivel bajo a nivel alto en un pin del microcontrolador, es posible que entre a esta rutina por alguna interferencia, por lo tanto para comprobar que si hay conexión física se vuelve a sensor el pin. Ya establecida la conexión física se procede a realizar la conexión lógica con el host (PC) y espera que desde el software del PC le envíe la opción de control.

Sí la opción es leer los datos, el microcontrolador recupera la última dirección de memoria utilizada, cuyo valor esta grabado en los dos primeros registros de la memoria externa y comienza a enviar los datos almacenados desde la tercera dirección de memoria hasta el último valor almacenado, la forma en que envía los datos al host es en grupos de 7 (código del lugar, día, mes, año, hora, minutos y segundos), además, envía los caracteres del formato de fecha (/) y de la hora (:). De esta manera en el PC se puede visualizar los datos organizados.

Figura 54. Datos recibidos por el PC



Una vez realizada la lectura pasa nuevamente a esperar la próxima opción que le asigne el host.

Con la opción de borrar la memoria, se inicializa el puntero al grabar en primer registro de la memoria externa con el valor de 0 y en el segundo registro el valor de 1, de esta forma cuando el programa principal llama a la función grabar, esta empieza a almacenar los datos recibidos desde la última dirección usada mas uno, la cual sería en este caso la dirección 2 (tercera posición de memoria). Al escoger esta opción los datos realmente no son borrados, simplemente se cambia la dirección de memoria al inicio para que los datos nuevos sobrescriban los anteriores. Al finalizar esta función, vuelve a esperar la nueva opción del host. Finalmente, la opción salir le permite terminar con esta interrupción y se puede desconectar físicamente el dispositivo móvil para continuar con su programa principal.

### 2.3.3.2 Configuración de registros del microcontrolador.

- **Configuración del oscilador.** La configuración del oscilador del microcontrolador PIC18F2550 se realiza por medio de los registros de configuración CONFIG1L y CONFIG1H y los registros de control OSCCON y OSTUNE.

El modo seleccionado es XT con un cristal de 4MHz y según el fabricante sugiere utilizar condensadores de 27pF para conectar dicho cristal con el microcontrolador.

Los modos de funcionamiento del oscilador del PIC18F2550 y el módulo USB están muy relacionados. En microcontroladores PIC anteriores a la familia PIC18F2455/2550/4455/4550, todos los módulos del microcontrolador, incluidos los puertos de comunicaciones, utilizan la misma fuente de reloj, pero en los microcontroladores de esta familia, el oscilador primario forma parte del módulo USB, de manera que este módulo sólo puede tomar la señal de reloj del oscilador primario. Sin embargo, el resto del micro puede utilizar cualquier otra fuente.

Dependiendo, si se utiliza USB de velocidad baja o completa, la señal de reloj del módulo USB deberá ser de 6 MHz o de 48 MHz, respectivamente. Para que el dispositivo sea compatible con la versión de USB 2.0 se debe escoger una frecuencia de 48MHz para USB.

Los registros de configuración, CONFIG1L y CONFIG1H, permiten seleccionar el modo de funcionamiento del oscilador y las opciones de pre y post-escalado. Para conseguir los 48MHz como fuente de reloj para el puerto USB se configura los registros como se muestra en el anexo I.

Tabla 23. Configuración del registro CONFIG1L

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
—	—	USBDIV	CPUDIV1	CPUDIV0	PLLDIV2	PLLDIV1	PLLDIV0
—	—	1	0	0	0	0	0

En PLLDIV (bits 2:0) del registro CONFIG1L se los configura con 0 para no hacer ninguna división y así la frecuencia de 4MHz del oscilador externo entra directamente en el PLL, el valor de USBDIV (bit 5) se lo establece en nivel alto para hacer la división del PLL el cual tiene una frecuencia de 96MHz, obteniendo de esta forma una frecuencia para el módulo de USB a 48MHz. En CPUDIV (bits 4 y 3) se les asigna el valor de 0 para que la frecuencia del oscilador primario (4MHz) pase igual a los demás módulos del microcontrolador.

Tabla 24. Configuración del registro CONFIG1H

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IESO	FCMEN	—	—	FOSC3	FOSC2	FOSC1	FOSC0
1	0	—	—	0	0	0	0

El bit IESO (bit 7) permite el intercambio del oscilador (interno/externo) pero como en este caso solo se utiliza el oscilador externo, se deshabilita esta opción colocándolo en bajo el bit 7, los bits de FOSC (bits 3:0) se los configura con 0 (oscilador XT), esta parte sirve para establecer definitivamente la frecuencia de reloj que va utilizar el microcontrolador.

El registro de control OSCTUNE sirve para calibrar y ajustar la frecuencia del oscilador interno pero debido a que únicamente se utiliza el oscilador primario como fuente del reloj, el valor de este registro no interesa.

Tabla 25. Configuración del registro OSCCON

R/W-0	R/W-1	R/W-0	R/W-0	R <sup>(1)</sup>	R-0	R/W-0	R/W-0
IDLEN	IRCF2	IRCF1	IRCF0	OSTS	IOFS	SCS1	SCS0
0	1	0	0			0	0
bit 7							bit 0

En el registro OSCCON se controla varios aspectos de la operación del reloj y una de las más importantes es la que selecciona la fuente de reloj que se va utilizar (bits SCS1 y SCS0) ya sea interna (10 ó 11), externa (01) o primaria (00).

• **Configuración del módulo USB.** El control y la configuración del módulo USB se lleva a cabo a través de 22 registros de estado y control. Estos son:

- Registro de control USB (UCON).
- Registro de configuración USB (UCFG).
- Registro de estado de transferencia USB (USTAT).
- Registro de dirección de dispositivo USB (UADDR).
- Registros de número de frame (UFRMH:UFRML).
- Registros (del 0 al 15) de habilitación de puntos finales (UEPn).

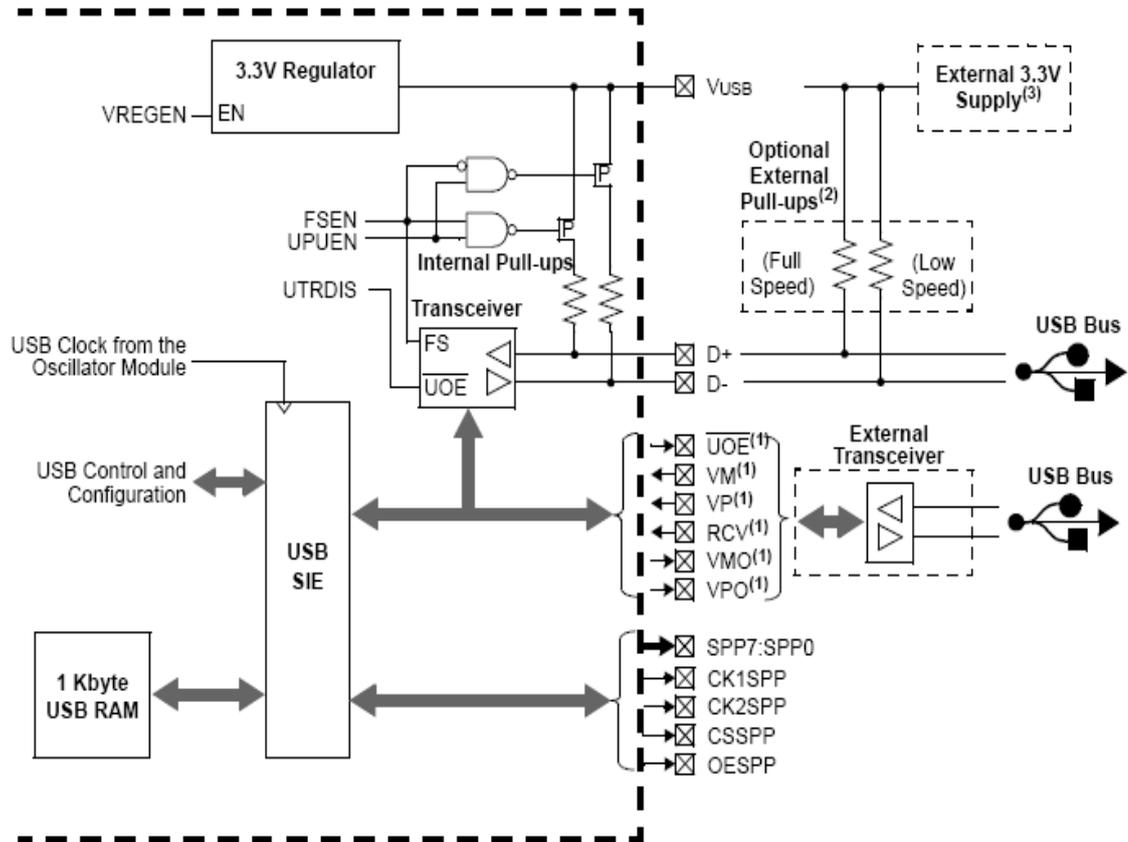
El registro de control UCON contiene los bits necesarios para controlar el comportamiento del módulo USB durante las transferencias.

Tabla 26. Configuración del registro UCON

U-0	R/W-0	R-x	R/C-0	R/W-0	R/W-0	R/W-0	U-0
—	PPBRST	SE0	PKTDIS	USBEN	RESUME	SUSPND	—
—	0	0	1	0	0	0	—
bit 7							bit 0

El módulo USB dispone de un regulador interno de 3,3 V para suministrar tensión al transceiver interno. Esto se debe a que USB utiliza 3,3 V para las comunicaciones, por lo tanto, mientras el resto del micro puede utilizar una alimentación de 5 V, el transceiver debe alimentarse de una fuente distinta (Vusb). El regulador se controla a través del bit VREGEN del registro de configuración CONFIG2L. Cuando está habilitado (VREGEN = 1), el voltaje es visible en el pin Vusb. Si el regulador está deshabilitado, una tensión de 3,3 V debe ser suministrada externamente a través de Vusb. Se debe tener en cuenta que las resistencias de pull-ups internas únicamente pueden utilizarse si el transceiver interno está activo.

Figura 55. Periférico USB y opciones



Fuente: PIC16F628A datasheet [Online]. Microchip Technology Inc.

La configuración del transceiver y los pull-ups así como de la velocidad del dispositivo se debe llevar a cabo antes de habilitar el módulo y no puede ser cambiada mientras esté habilitado. El registro que se encarga de realizar esta configuración es UCFG.

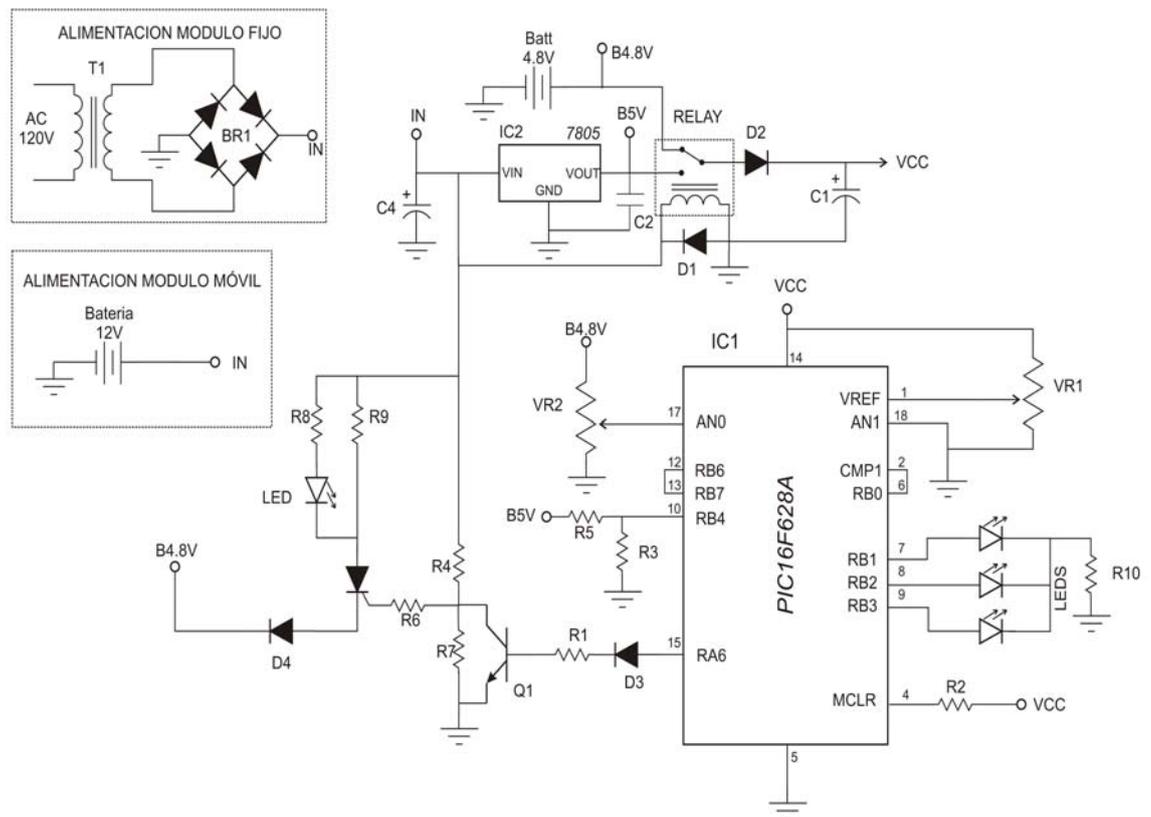
Tabla 27. Configuración del registro UCFG

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
UTEYE	UOEMON	—	UPUEN	UTRDIS	FSEN	PPB1	PPB0
0	0	—	1	0	1	0	0
bit 7							bit 0

El bit UPUEN (bit 4) del registro UCFG permite activar las resistencias pull-ups internas para el manejo de la velocidad de transferencia de datos y con el bit FSEN (bit 2) se selecciona una de las dos resistencias: con 1 se activa la resistencia que esta conectada al pin D+ del puerto USB para trabajar con velocidad completa; con 0 selecciona la resistencia conectada a D- para trabajar con velocidad baja.

### 2.3.4 Programación y configuración microcontrolador del sistema de alimentación.

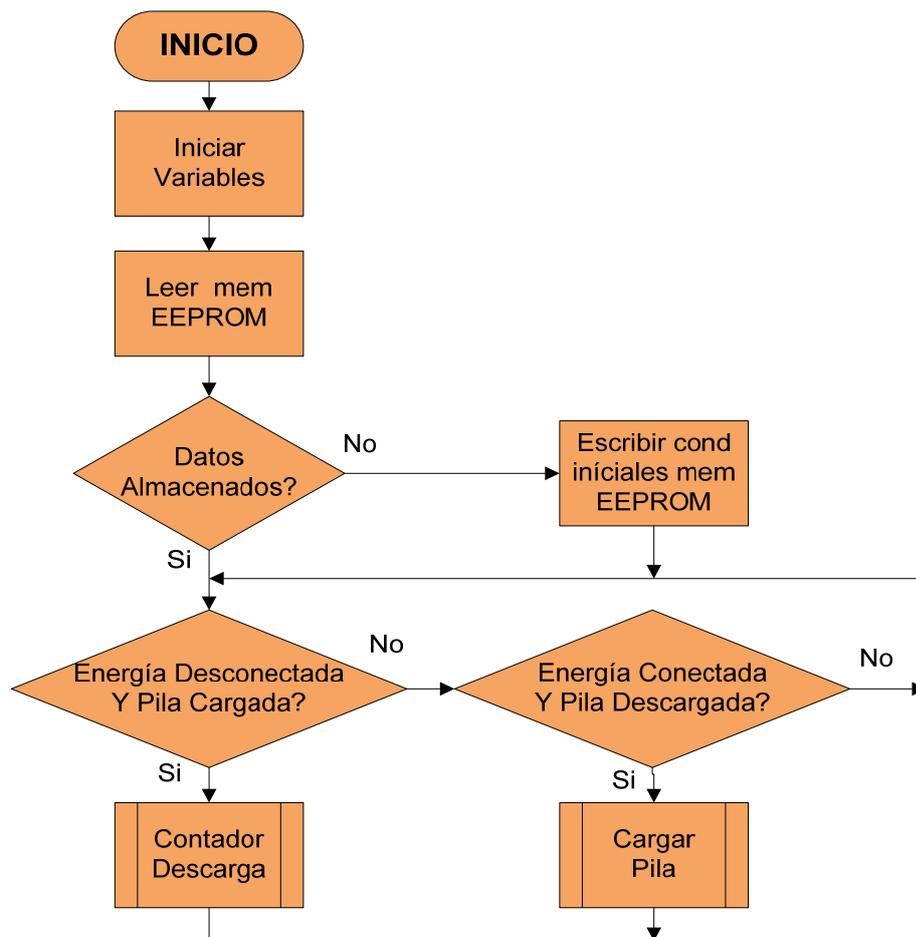
Figura 56. Diagrama eléctrico UPS



**2.3.4.1 Desarrollo de software implementado en microcontrolador.** La fuente de alimentación cuenta con un microcontrolador PIC16F628A, el cual se programa para realizar el monitoreo de carga de los acumuladores y generar los pulsos para la recarga de estos.

- **Rutina principal.**

Figura 57. Diagrama de flujo rutina principal microcontrolador de UPS



En el proceso de inicializar variables, se reinicia los contadores utilizados para medir el tiempo de carga y descarga de las baterías recargables.

El pin RB4 (control de carga) del microcontrolador es programado como entrada a la cual se conecta la fuente principal de voltaje, este permite sensar la presencia de la alimentación principal. El nivel de los acumuladores (cargado o descargado) depende del valor de los contadores de descarga almacenados en la memoria EEPROM

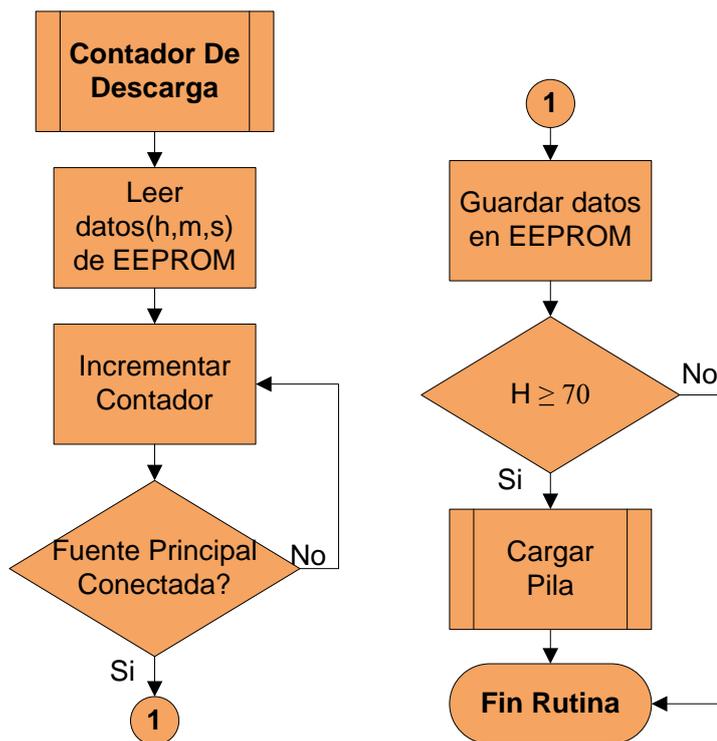
Figura 58. Mapa de memoria EEPROM PIC16F628A

PIC CPU EPROM Memory - PIC16F628A UPS																			
00	FF	FF	FF	00	00	00	FF	00	00	00	FF								
10	FF																		
20	FF																		
30	FF																		
40	FF																		
50	FF																		
60	FF																		
70	FF																		

Dependiendo del nivel de carga de los acumuladores, y de la conexión o desconexión de la fuente principal, el microcontrolador entrará a la subrutina de contador de descarga o a la subrutina de cargar baterías.

- **Subrutina contador de descarga.**

Figura 59. Diagrama de flujo subrutina contador de descarga



En el momento en que se presente desconexión de la fuente principal se hace la conmutación automática a las baterías de respaldo, el microcontrolador detecta la ausencia de voltaje principal y verifica el nivel de los acumuladores si estos se encuentran cargados el microcontrolador entra a la subrutina de contador de descarga en la cual se cuenta el tiempo de ausencia de la alimentación principal.

Los valores de las variables temporales del contador de descarga (horas, minutos, segundos) se les asigna el valor almacenado en la memoria EEPROM (los datos están ubicados desde la dirección 0x03 hasta 0x05 respectivamente, ver figura 58), estas variables se incrementan hasta que la fuente principal de voltaje se conecte, en ese momento se guardan los datos de los contadores en la memoria EEPROM.

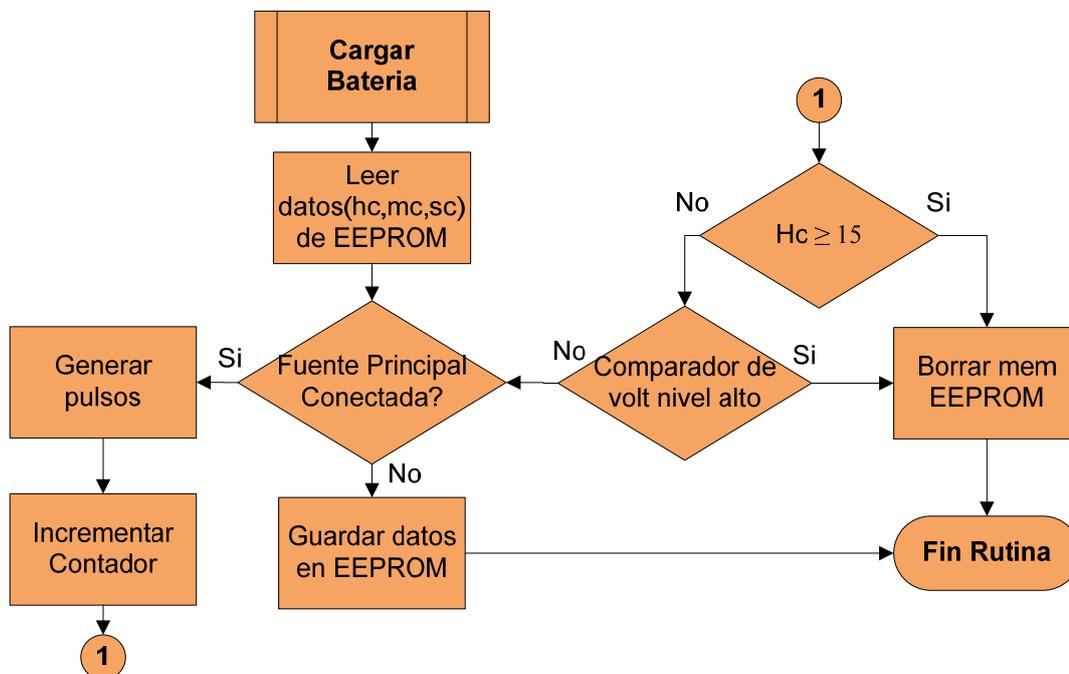
El nivel de carga de los acumuladores depende del tiempo en horas que estuvieron alimentando al circuito principal ya sea del módulo fijo, móvil o de configuración, cada uno de los módulos tiene un consumo de corriente en mA (miliamperios hora) diferente, esto hace que la duración de carga de los acumuladores varíe de acuerdo al módulo que se alimente.

Por ejemplo, el módulo fijo tiene un consumo de 13 mA y los acumuladores conectados del sistema son de 2700 mA, lo que permite tener una autonomía de funcionamiento de 200 horas, sin conexión a la fuente principal de voltaje que en este caso es tomada de la red eléctrica.

Para recargar la batería se determina que el nivel de esta debe estar en un 50%, por lo que el contador de horas de descarga para este módulo debe alcanzar un valor igual o superior a 100.

- **Subrutina cargar batería.**

Figura 60. Diagrama de flujo subrutina cargar batería



Cuando el valor de horas del contador de descarga alcanza el valor fijado y la fuente principal alimenta al módulo, el microcontrolador del sistema de alimentación procesa la subrutina de carga de acumuladores.

A las variables del contador de carga se les asigna el valor almacenado en la memoria EEPROM (los datos están ubicados desde la dirección 0x07 hasta 0x09 horas, minutos, segundos respectivamente, ver figura 58), este contador se incrementa hasta que alcance el número de horas estimado para la recarga. Durante este tiempo, el microcontrolador activa una salida que genera pulsos con una frecuencia de 120Hz los que hacen la activación o desactivación del circuito electrónico que permite cargar las baterías.

Si durante el proceso de la recarga se presenta la desconexión de la fuente principal, se desactivan los pulsos y los datos del contador (horas, minutos, segundos) se almacenan en la memoria EEPROM, cuando se conecte nuevamente la alimentación de la fuente principal, se lee los valores de los contadores y se reinicia la recarga hasta que se alcance el tiempo estipulado para ello.

El circuito electrónico de carga hace circular por los acumuladores una corriente pulsante de 180mA, como se menciona anteriormente las baterías tienen una capacidad de 2700mAH, esto implica que los acumuladores necesitan 15 horas para almacenar 100% de carga.

El tiempo estimado para cargar la baterías se programa en 10 horas debido a que la recarga se inicia cuando el nivel de carga de los acumuladores esté en 50%.

A medida que los acumuladores pierden su carga, el voltaje que suministran también se decrementa, de la misma manera, si se los esta recargando empiezan a acumular carga y el voltaje generado empieza a incrementarse hasta que llegar a un valor máximo en el que se estabiliza, si sigue suministrando corriente de carga los acumuladores aumentan su temperatura lo que hace que pierdan su vida útil.

Además de la lógica de temporización se configura el módulo de comparadores de entradas análogas del microcontrolador. El comparador se activa cuando el voltaje de las baterías alcanza su máximo valor, esto hace que el microcontrolador finalice la rutina de carga reiniciando los valores de los contadores de carga y descarga almacenados en la memoria EEPROM. La desconexión del circuito de recarga se realiza ya sea por el temporizador o por la activación del comparador.

**2.3.4.2 Configuración de registros del microcontrolador.** A continuación se describe los registros y parámetros configurados en el microcontrolador PIC16F628A según los módulos utilizados.

- **Configuración del oscilador.** El modo seleccionado es el INTOSC, oscilador interno, configurando con los bits FOSC2 = 1, FOSC1 = 0, FOSC0 = 0.

Tabla 28. Configuración del registro CONFIG

$\overline{CP}$	—	—	—	—	$\overline{CPD}$	LVP	BOREN	MCLRRE	FOSC2	$\overline{PWRTÉ}$	WDTE	FOSC1	FOSC0	
0					0	0	0	1	1	0	0	0	0	
bit 13														bit 0

Al seleccionar el modo de oscilación interno se escoge la frecuencia del bus de datos y señales de reloj, configurando el registro PCON (registro de control de frecuencia).

Tabla 29. Configuración del registro PCON

—	—	—	—	OSCF	—	$\overline{POR}$	$\overline{BOR}$
—	—	—	—	1	—	1	1
bit 7				bit 0			

El bit OSCF permite determinar la frecuencia del microcontrolador cuando se configura para que trabaje con el oscilador interno, OSCF = 1 determina la frecuencia de oscilación en 4Mhz y OSCF = 0 en 48 Khz.

- **Configuración módulo comparadores.** Contiene dos comparadores análogos, las entradas están multiplexadas en los pines RA0 a RA3.

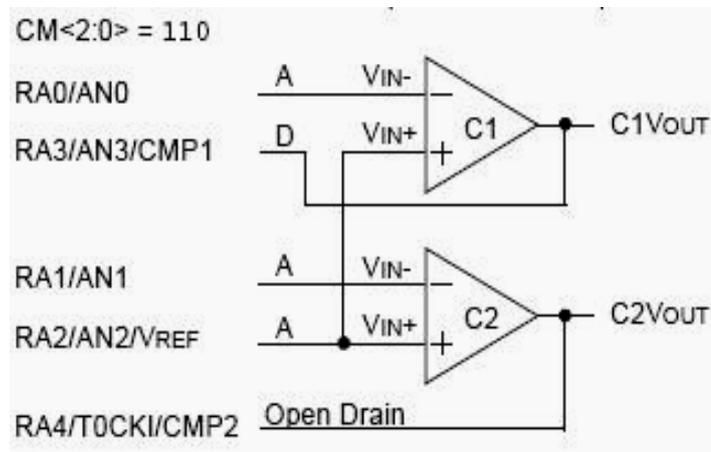
El registro CMCOM (registro de configuración del comparador), permite definir las entradas y salidas de los comparadores.

Tabla 30. Configuración del registro CMCOM

C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
0	0	0	0	0	1	1	0
bit 7				bit 0			

Los bits CM0 = 0, CM1 = 1 Y CM2 = 1, seleccionan el modo o conexión interna del comparador como se indica en la siguiente figura.

Figura 61. Conexión interna de módulo de comparadores



Fuente: PIC16F628A datasheet [Online]. Microchip Technology Inc.

Los bits C1INV Y C2INV configuran la salida de comparadores: invertida (bits = 1) o no invertida (bits = 0).

## 2.4 FIRMWARE, DRIVER, SOFTWARE Y PROCEDIMIENTOS DE INSTALACIÓN.

Se implementa una comunicación bidireccional serial entre el módulo móvil y el PC el cual utiliza un puerto COMM serie estándar pero emulado vía USB 2.0 a velocidad completa. Para ello se realiza los ajustes necesarios en el Firmware del PIC18F2550 para que sea reconocido por el Windows como un dispositivo serie estándar y lo registre como un puerto COMM virtual. Además, se utiliza el driver *mchpcdc.inf* desarrollado por Microchip para Windows XP y Vista.

Para el desarrollo del firmware se modifican algunas características de los archivos que proporciona el compilador y que se utilizan en el proyecto. Estos archivos contienen la definición de los registros y funciones USB. Estos ficheros, son:

**18F2550.h:** Librería de cabecera estándar para dispositivos PIC18F2550.

**pic18\_usb.h:** Capa hardware de los dispositivos PIC18Fxx5x.

**USB.C:** Manejador de peticiones USB estándar.

**USB.H:** Prototipos de funciones, definiciones y variables globales USB.

**usb\_cdc.h:** Librería para adicionar un puerto virtual en el PC a través de USB usando el CDC (Communication Device Class).

**USB\_ConBuses.h:** Configuración y descriptores de dispositivos USB.

En la librería *USB\_ConBuses.h*, se definen las estructuras y parámetros necesarios para la conexión USB, necesarios para una correcta conexión con el driver del PC. De esta librería se modifica el *VID&PID* y la tabla *USB\_STRING\_DESC[]*.

El **VID** (Identificación del Vendedor), es un número de 16 bits, este código identifica al fabricante del hardware a conectar. En este caso se utiliza el número 04D8h que identifica a Microchip.

El **PID** (Identificación del Producto), es un número de 16 bits, este código identifica al dispositivo en concreto (hardware a conectar). Para el microcontrolador 18F2550 se utiliza el número 000Bh que identifica a la familia de los PIC18 de este fabricante.

Al conectar el dispositivo móvil al PC, el sistema operativo de este (Windows XP) se conecta con el firmware, el S.O. recibirá el VID&PID y busca entre sus drivers instalados para encontrar el que corresponde a esta identificación. Como este no corresponde a un dispositivo genérico se debe indicar la ubicación del driver adecuado. Este driver debe estar configurado para conectar con un hardware cuyo VID&PID sea el mismo. Como el driver para puertos serie ya existe en Windows, se instala el driver *mchpcdc.inf* para que el S.O se conecte correctamente con el firmware que tiene el PIC.

La tabla *USB\_STRING\_DESC* de la librería *USB\_ConBuses.h*, contiene la descripción del dispositivo detectado por el driver de Windows XP y permite mostrar la descripción del dispositivo en la lista del hardware instalado en el sistema. Esta se divide en dos partes o tablas, la *USB\_STRING\_DESC* que contiene las descripciones requeridas y una tabla accesoria llamada *USB\_STRING\_DESC\_OFFSET* que contiene los offset, o desplazamientos con respecto al inicio de *USB\_STRING\_DESC* en donde se encuentran las correspondientes cadenas. Cada una de ellas consta de tres elementos.

*USB\_STRING\_DESC\_OFFSET* tiene tres números que indican donde comienza el correspondiente dato en la tabla *USB\_STRING\_DESC*. Así un contenido de {0, 4, 12} indica que el primer string comienza en el byte 0, el segundo en el byte 4 y el tercero se encuentra a partir del byte número 12. Si se cambia la longitud de cualquiera de los strings, se debe reordenar esta tabla correspondientemente con solo contar los caracteres y apuntar en esta tabla el número de byte donde comienza cada uno de ellos.

*USB\_STRING\_DESC* contiene los tres strings que se va a transmitir con el descriptor USB. Cada uno de ellos tiene la misma estructura que consta de un primer byte que indica la longitud total de la correspondiente cadena, un segundo byte que indica el tipo de dato que viene a continuación y por último tantos bytes como sean necesarios en el contenido del string.

El primer dato de esta tabla es:

4, *USB\_DESC\_STRING\_TYPE*, 0x09, 0x04: el numero 4 indica el tamaño de la cadena en bytes. *USB\_DESC\_STRING\_TYPE*, identifica al tipo de descriptor (STRING = 0x03), indica que el tipo de dato que viene a continuación es un string.

Los siguientes datos son los dos strings que definen el dispositivo y cuya estructura es idéntica al caso anterior:

8, *USB\_DESC\_STRING\_TYPE*, 'R', 0, 'J', 0, 'C', 0: define el string con el nombre del fabricante "RJC".

22, *USB\_DESC\_STRING\_TYPE*, 'C', 0, 'o', 0, 'n', 0, 't', 0, 'r', 0, 'o', 0, 'l', 0, ' ', 0, 'D', 0, 'e', 0, ' ', 0, 'B', 0, 'u', 0, 's', 0, 'e', 0, 's', 0, en este string se define el nombre del dispositivo como "Control De Buses".

La modificación de la tabla *USB\_STRING\_DESC*, en la librería queda de la siguiente manera:

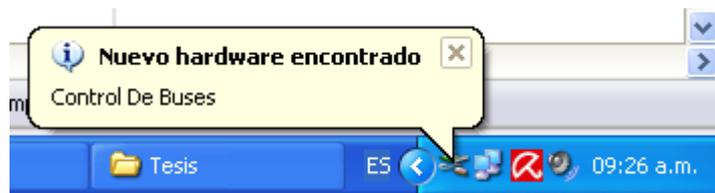
```
/*
*****
Descriptores de cadena. La cadena 0 es una cadena especial y debe incluirse
siempre
*****
//the offset of the starting location of each string.  offset[0] is the start of string 0,
offset[1] is the start of string 1, etc.
char USB_STRING_DESC_OFFSET[]={0,4,12};

char const USB_STRING_DESC[]={
    //string 0
    4, //length of string index
    USB_DESC_STRING_TYPE, //descriptor type 0x03 (STRING)
    0x09,0x04, //Microsoft Defined for US-English
    //string 1
    8, //length of string index
    USB_DESC_STRING_TYPE, //descriptor type 0x03 (STRING)
    'R',0,
    'J',0,
    'C',0,
    //string 2 --> nombre del dispositivo
    34, //length of string index
    USB_DESC_STRING_TYPE, //descriptor type 0x03 (STRING)
    'C',0, 'o',0, 'n',0, 't',0, 'r',0, 'o',0, 'l',0, ' ',0, 'D',0, 'e',0, ' ',0,
    'B',0, 'u',0, 's',0, 'e',0, 's',0
};
```

El contenido completo de la librería se muestra en el anexo O.

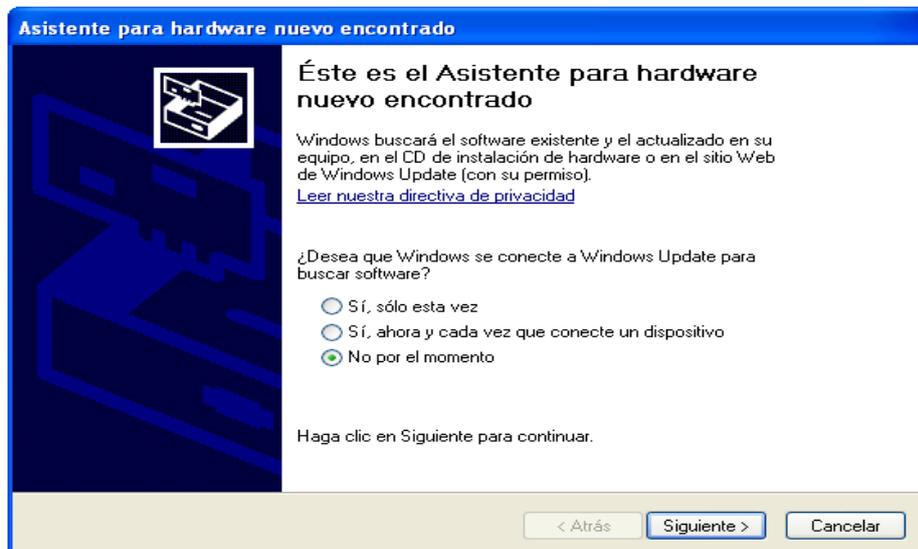
**2.4.1 Instalación del dispositivo móvil.** Al conectar el dispositivo móvil al PC por primera vez, aparece el asistente de Windows para la instalación de nuevo hardware:

Figura 62. Instalación de nuevo hardware (a)



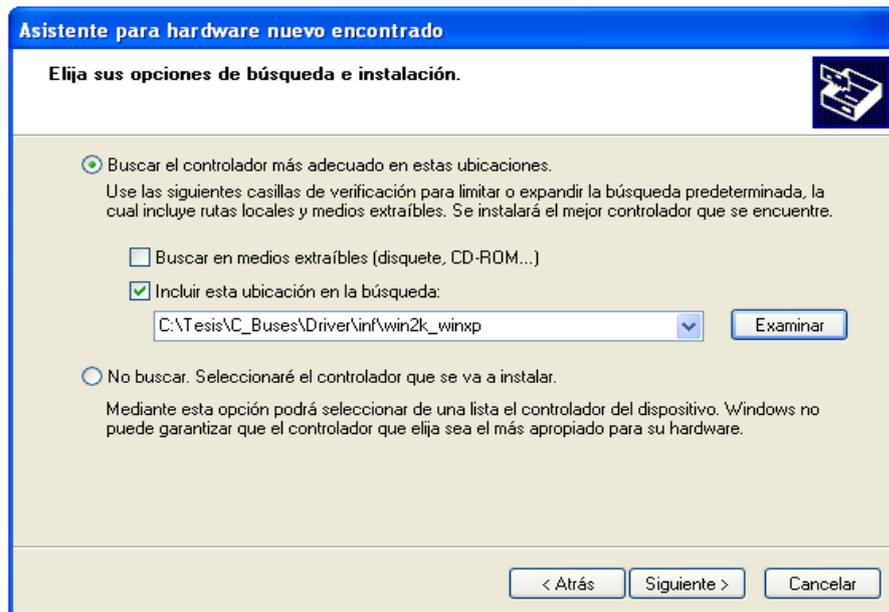
Automáticamente pregunta si se quiere conectar a Windows Update. Se marca la opción *No* y se da clic sobre *Siguiente*:

Figura 63. Instalación de nuevo hardware (b)



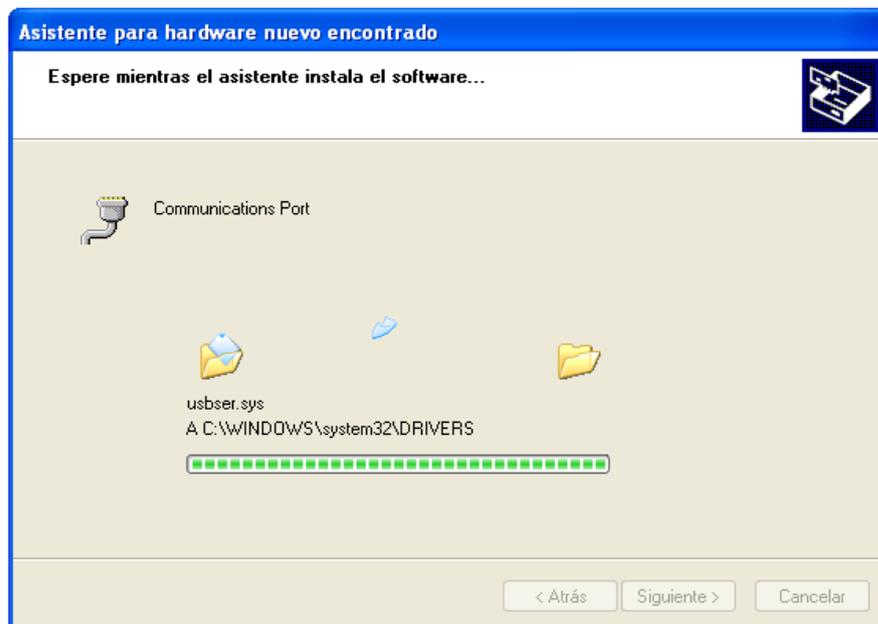
En la siguiente pantalla se marca la opción de "*Instalar desde una lista o ubicación específica*". Se da clic sobre *Siguiente*: a la próxima pantalla, donde se marca la opción de "*Buscar el controlador más adecuado en estas ubicaciones*". Se presiona sobre "*Incluir esta ubicación en la búsqueda*" y a continuación sobre *Examinar*, se selecciona la ubicación de la carpeta que contiene el driver *mchpcdc.inf*, y se le da clic en *Aceptar* y en *Siguiente*.

Figura 64. Instalación de nuevo hardware (c)



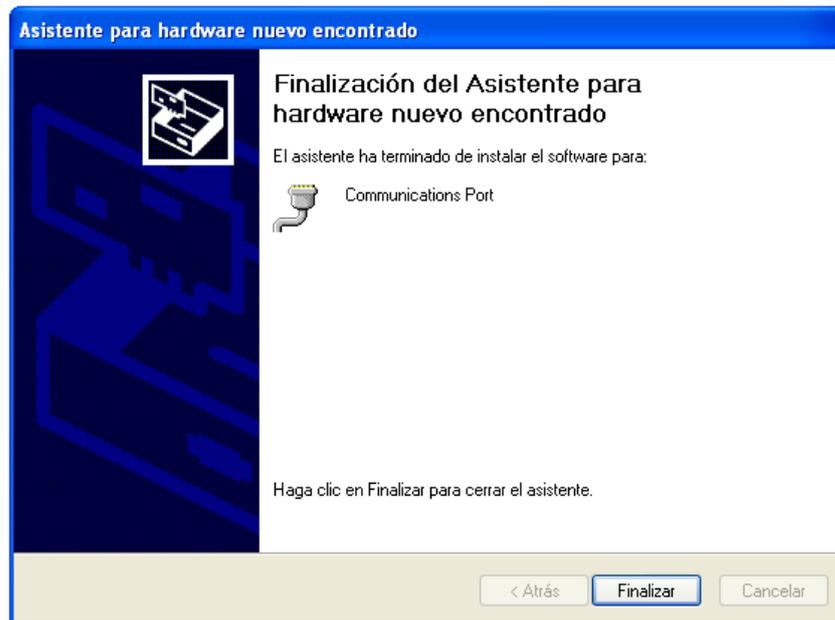
En ese momento, comienza a copiarse los archivos necesarios en el computador.

Figura 65. Instalación de nuevo hardware (d)



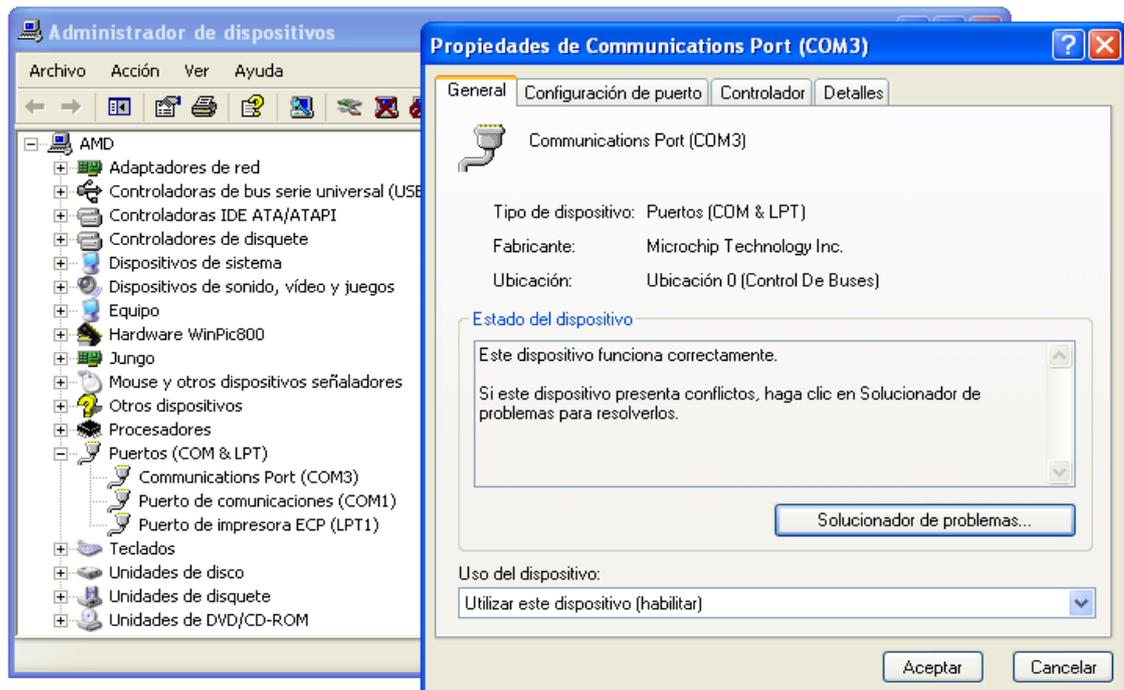
Al terminar de copiar, aparece una pantalla indicando que la instalación del driver está finalizada.

Figura 66. Instalación de nuevo hardware (e)



Si la instalación del dispositivo que se realiza no tuvo ningún inconveniente, en el Administrador de dispositivos debe aparecer un puerto COM virtual nuevo.

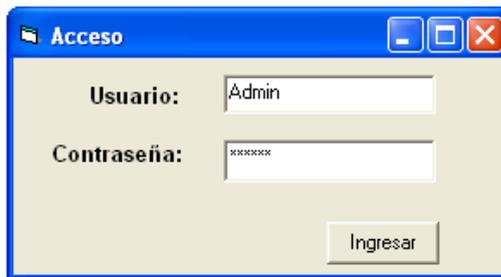
Figura 67. Instalación de nuevo hardware (f)



En las siguientes ocasiones que se conecte el dispositivo al computador, este es reconocido de forma automática, ya no es necesario indicar ningún tipo de información.

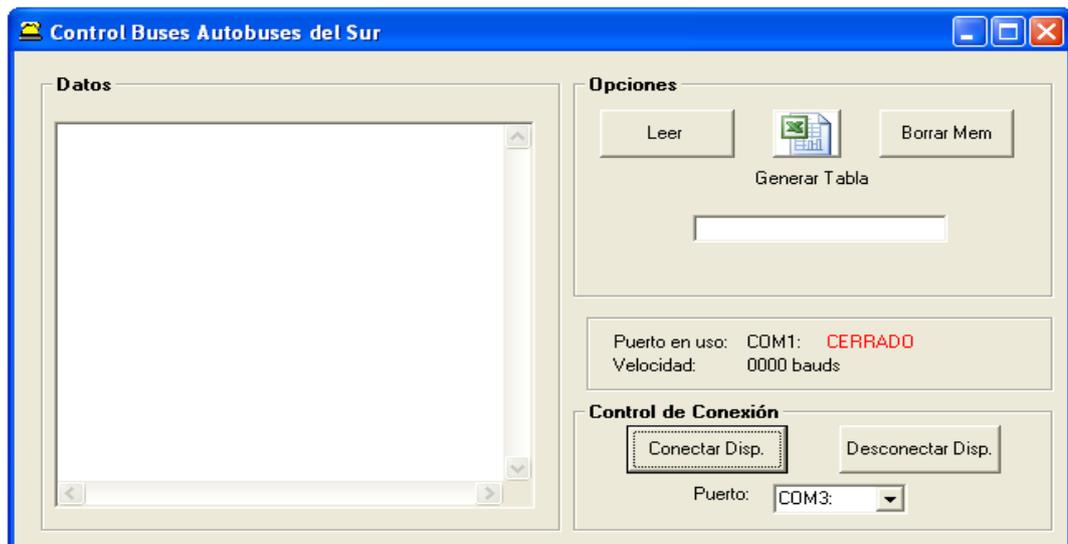
**2.4.2 Desarrollo de la aplicación.** Para poder visualizar en PC los datos recolectados por el módulo móvil se implementa una pequeña aplicación desarrollada en Visual Basic 6, dicha aplicación se compila con el nombre de *Control Buses.exe*. A la aplicación se le incluye una etapa de seguridad para iniciar el programa, la cual solicita el nombre de usuario y contraseña, evitando así que personas no autorizadas tengan acceso a los datos de los módulos.

Figura 68. Seguridad de acceso



Una vez validado el usuario, se muestra en pantalla la aplicación principal la cual permite gestionar la conexión entre el módulo móvil y el PC, se puede visualizar, grabar los datos e inicializar la memoria del dispositivo.

Figura 69. Aplicación principal para el control de buses



Como la comunicación USB entre el PC y el módulo móvil se realiza mediante un puerto virtual COMM, el usuario debe escoger número del puerto que se le asigna en la instalación del driver (ver figura 67), posteriormente, se da un click en el botón *Conectar Disp* para establecer la interconexión cliente-función, que permite definir la velocidad de transmisión y el protocolo RS-232.

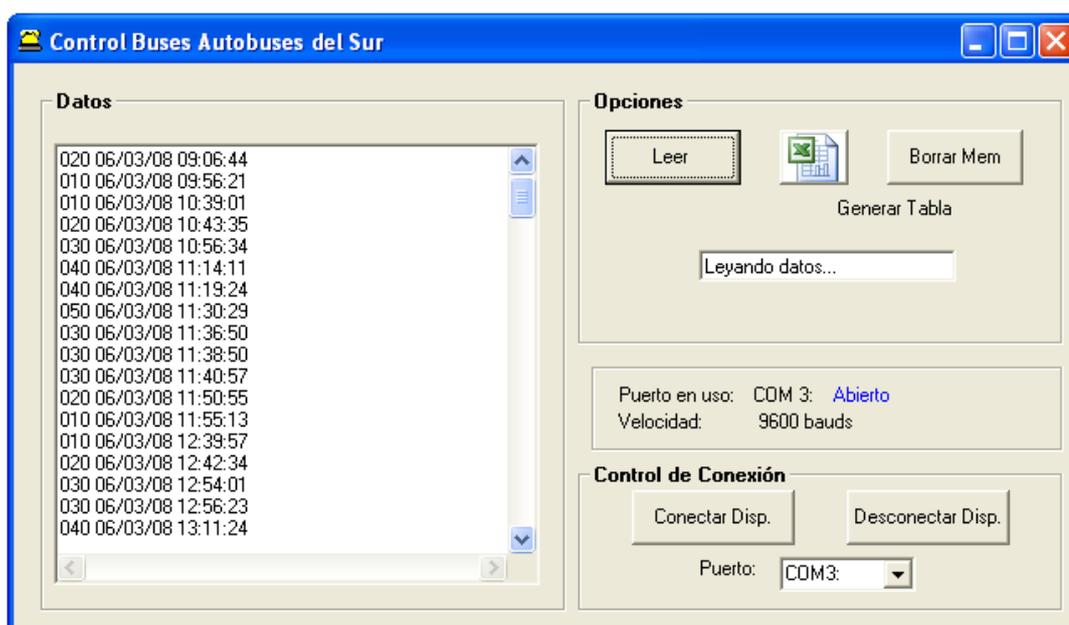
En el recuadro *Opciones* se tiene tres botones que permiten la manipulación de la información.

Figura 70. Ventana control de datos



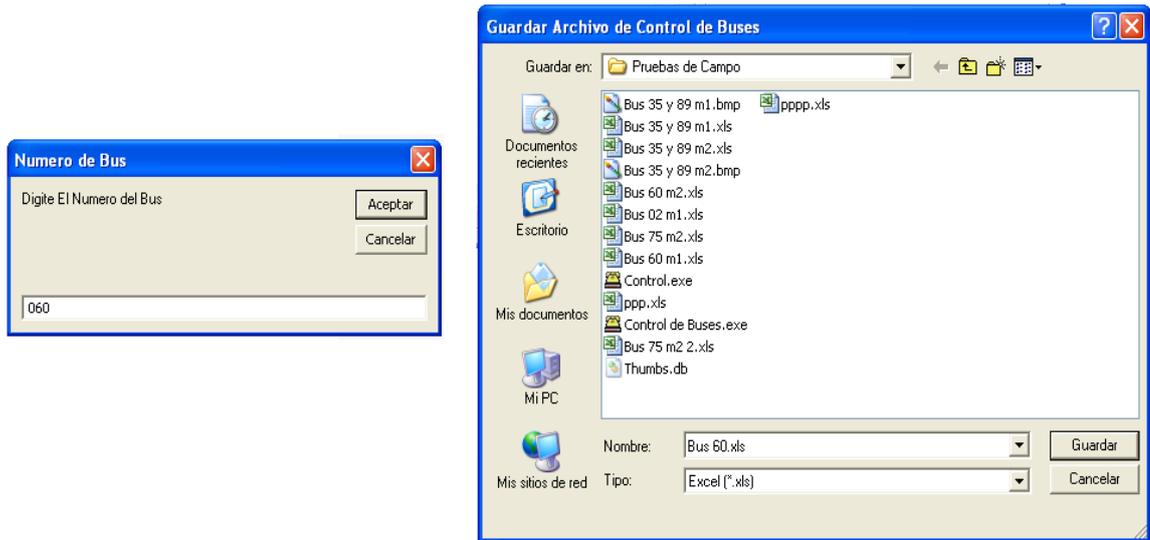
- El botón Leer se implementa para recuperar los datos almacenados en la memoria EEPROM del dispositivo móvil, que se visualizan en la sub-ventana *Datos*.

Figura 71. Visualización de datos



- El botón *Generar Tabla* permite realizar la exportación de los datos leídos a un archivo de Excel, en este proceso se pide inicialmente el ingreso del número del vehículo portador del dispositivo móvil, además se debe ingresar la dirección de la carpeta en la cual se va almacenar y el nombre que se le va a dar al archivo XLS.

Figura 72. Guardar datos en el PC



Después de almacenar el archivo la información se visualiza en una tabla en Microsoft Excel.

Figura 73. Visualización datos en Microsoft Excel

	A	B	C
1	Bus 75	Fecha: 11/03/2008	
2	LUGAR	FECHA	HORA
3			
4	20	03/06/2008	09:06:44
5	10	03/06/2008	09:56:21
6	10	03/06/2008	10:39:01
7	20	03/06/2008	10:43:35
8	30	03/06/2008	10:56:34
9	40	03/06/2008	11:14:11
10	40	03/06/2008	11:19:24
11	50	03/06/2008	11:30:29
12	30	03/06/2008	11:36:50
13	30	03/06/2008	11:38:50
14	30	03/06/2008	11:40:57
15	20	03/06/2008	11:50:55
16	10	03/06/2008	11:55:13

- El botón *Borrar Mem* envía un código al microcontrolador para inicializar el puntero del manejo de la memoria EEPROM externa, permitiendo así sobrescribirla con los nuevos datos.

Para realizar la desconexión del dispositivo se hace un click en el botón *Desconectar Disp*, esta opción envía un dato al microcontrolador para finalizar la comunicación USB y así el módulo móvil puede continuar con su función principal (recolectar datos).

### 3. PRUEBAS E IMPLEMENTACIÓN

Una vez finalizada la fase de diseño del sistema de control de buses, se procede a la etapa de prueba, la cual comprende: la revisión y funcionamiento de los componentes electrónicos, la programación de los microcontroladores, la comunicación RF entre módulo fijo y móvil, la interfaz USB entre el módulo móvil y el computador.

Posteriormente se procede al ensamble de cada módulo del sistema.

Figura 74. Vista interior módulo fijo

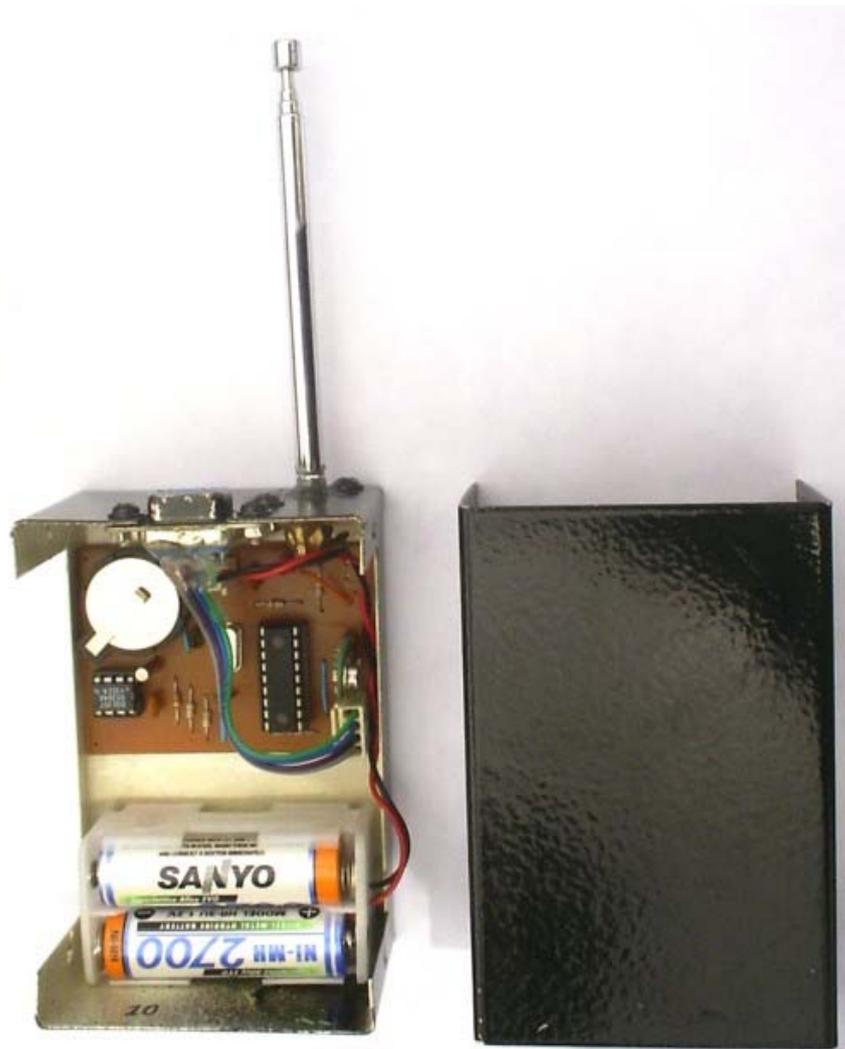


Figura 75. Vista interior módulo móvil

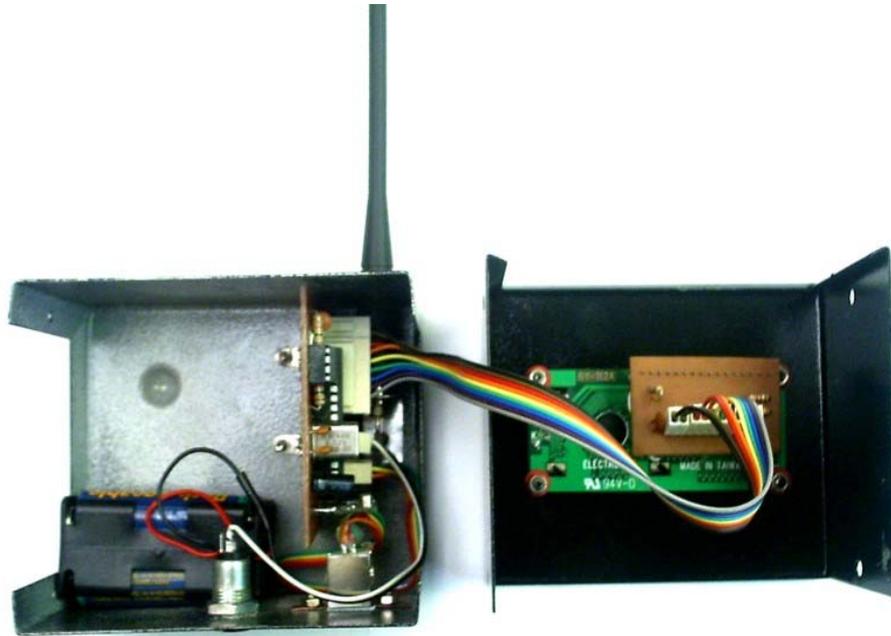


Figura 76. Vista exterior módulo móvil



Figura 77. Vista interior módulo de configuración

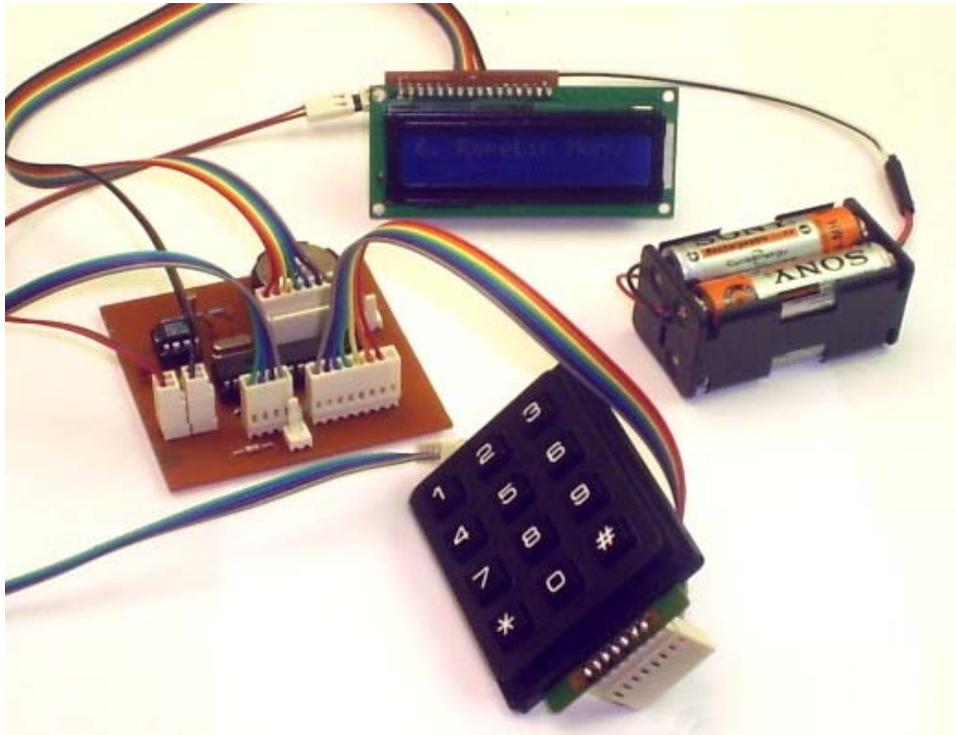


Figura 78. Vista exterior módulo de configuración



Para evaluar el correcto funcionamiento del sistema desarrollado y de la aplicación implementada, se solicita a la **Empresa Transportadora Autobuses Del Sur**, permiso para instalar en dos de sus vehículos el dispositivo móvil (ver anexo J), además, se ubican cinco módulos fijos, distribuidos en los puestos de control de tiempo que realiza la empresa para la ruta nº 12. Esto permite probar los módulos en un ambiente real de trabajo con el fin de hacer ajustes y correcciones en los mismos.

Los puntos de control donde se ubican los módulos fijos, son: El Rosario (figura 79), Miraflores (figura 80), Avenida las Américas (figura 81), UDENAR (figura 82) y La Panadería (figura 83):

Figura 79. Barrió El Rosario (diagonal 16, cra 13 este), código 10



Figura 80. Barrió Miraflores (calle 16D, Cra 2E), código 20



Figura 81. Avenida las Américas (Av Américas, calle 13), código 30



Figura 82. Universidad de Nariño (calle 18, cra 50), código 40



Figura 83. Iglesia San Sebastián (calle 21, cra 21), código 50



Las pruebas se dividen, en:

- Pruebas preliminares
- Corrección de errores y pruebas finales

### **3.1 PRUEBAS PRELIMINARES.**

Con los módulos fijos y móviles ubicados en su respectivo sitio se realizan estas pruebas con el propósito de evaluar el comportamiento de los programas y de los dispositivos electrónicos de cada uno de los módulos, en 8 días de recolección de datos en el módulo móvil se encuentra que almacena algunos datos erróneos en la memoria EEPROM como se muestra en la figura 84, se comprueba que se debe al ruido e interferencia generada por dispositivos electrónicos de RF como celulares.

Figura 84. Datos recolectados en las primeras pruebas

	A	B	C
1	Control Bus 02	Fecha: 29/02/2008	
2	20	28/02/08	17:41:09
3	30	29/02/08	07:57:17
4	30	29/02/08	08:35:03
5	20	29/02/08	08:41:03
6	50	29/02/08	08:51:21
7	40	29/02/08	09:05:58
8	40	29/02/08	09:14:02
9	10	29/02/08	-7:27:54
10	50	29/02/08	09:32:59
11	20	29/02/08	09:46:55
12	30	29/02/08	09:51:56
13	30	29/02/08	10:46:30
14	50	29/02/08	11:03:08
15	40	29/02/08	11:19:38
16	40	29/02/08	11:26:25
17	40	29/02/08	11:28:29
18	10	29/02/08	11:40:05
19	50	29/02/08	11:45:16
20	20	29/02/08	11:59:32
21	30	29/02/08	12:03:59
22	30	29/02/08	12:50:05
23	20	29/02/08	12:55:14
24	50	29/02/08	13:08:31
25	40	29/02/08	13:23:06
26	40	29/02/08	13:31:38
27	10	29/02/08	13:44:06
28	50	29/02/08	13:48:58
29	20	29/02/08	14:01:17
30	30	29/02/08	-0:55:18
31	30	29/02/08	14:57:21

### 3.2 CORRECCION DE ERRORES Y PRUEBAS FINALES.

Para resolver el problema anterior se modifica las rutinas de transmisión y recepción de los módulos, incluyendo dos bytes en la trama para corrección de errores, lo que permite mejorar la validación de los datos como se mencionó en la sección 2.3.1.1 desarrollo software (rutina de Transmisión), inicialmente para corrección de errores únicamente se verifica un byte de inicio y otro de fin de trama.

En la parte física de los dispositivos se aterriza el chasis (conexión a polo negativo) para crear un blindaje y así reducir las interferencias externas de radio frecuencia que afecta a los componentes electrónicos.

Después de realizar las respectivas modificaciones tanto de software como de hardware, se procede a instalar nuevamente los dispositivos para realizar las pruebas necesarias y garantizar el correcto funcionamiento del sistema.

La recolección de datos se lleva a cabo durante 15 días, periodo en el que no se presentan errores como en las pruebas preliminares.

Figura 85. Datos recolectados en pruebas finales

	A	B	C
1	Control Bus 60	Fecha:10/03/2008	
2	10	03/06/2008	11:38:55
3	20	03/06/2008	11:44:10
4	30	03/06/2008	11:53:44
5	40	03/06/2008	12:11:17
6	40	03/06/2008	12:17:45
7	50	03/06/2008	12:33:52
8	30	03/06/2008	12:38:04
9	30	03/06/2008	12:40:04
10	20	03/06/2008	12:52:44
11	10	03/06/2008	12:56:45
12	10	03/06/2008	12:59:10
13	10	03/06/2008	13:28:52
14	10	03/06/2008	13:30:53
15	20	03/06/2008	13:34:23
16	20	03/07/2008	05:58:28
17	10	03/07/2008	06:01:06
18	10	03/07/2008	06:24:16
19	10	03/07/2008	06:26:16
20	20	03/07/2008	06:30:52
21	20	03/07/2008	06:32:52
22	30	03/07/2008	06:42:32
23	40	03/07/2008	06:54:58
24	40	03/07/2008	07:04:40
25	50	03/07/2008	07:19:15
26	30	03/07/2008	07:22:40
27	30	03/07/2008	07:24:40
28	30	03/07/2008	07:26:41
29	20	03/07/2008	07:37:14
30	10	03/07/2008	07:40:27
31	10	03/07/2008	07:42:28

	A	B	C
1	Bus 75	Fecha:06/03/2008	
2	10	03/06/2008	10:39:01
3	20	03/06/2008	10:43:35
4	30	03/06/2008	10:56:34
5	40	03/06/2008	11:14:11
6	40	03/06/2008	11:19:24
7	50	03/06/2008	11:30:29
8	30	03/06/2008	11:36:50
9	30	03/06/2008	11:38:50
10	20	03/06/2008	11:50:55
11	10	03/06/2008	11:55:13
12	10	03/06/2008	12:39:57
13	20	03/06/2008	12:42:34
14	30	03/06/2008	12:54:01
15	40	03/06/2008	13:11:24
16	40	03/06/2008	13:18:03
17	50	03/06/2008	13:30:02
18	30	03/06/2008	13:36:46
19	30	03/06/2008	13:38:46
20	20	03/06/2008	13:51:26
21	10	03/06/2008	13:55:53
22	10	03/06/2008	14:40:19
23	20	03/06/2008	14:43:38
24	30	03/06/2008	14:54:23
25	30	03/06/2008	14:56:23
26	40	03/06/2008	15:16:18
27	40	03/06/2008	15:19:14
28	50	03/06/2008	15:30:48
29	30	03/06/2008	15:37:00
30	20	03/06/2008	15:51:33
31	10	03/06/2008	15:56:27

Los datos registrados se compararan con la hoja de control del bus correspondiente que la empresa maneja: por ejemplo para el bus N° 75 se obtuvieron los siguientes datos:

Figura 86. Tarjeta de control manual de tiempos bus 75

RELOJ	Orig.	RS	MIR	AV	UNI	RELOJ	RELOJ	UNI	AV	RS	Dest.	RELOJ
627	32	26	48	08		17	24	31	47	237		145
528	33	37	48	08		17	25	32	48	952		148
1092	37	41	53	08		17	29	36	52	1152		151
1232	37	41	53	08	12:54	17	29	36	52	1152		151
232	37	41	53	08	12:54	17	29	36	52	1152		151
652	57	01	13	28	15:51	37	49	56	12	817		51
652	57	01	13	28		37	49	56	12	817		51

Figura 87. Datos recolectados bus 75

ordenado Bus 75 m2 2.xls					
1	A	B	C	D	E
2	Bus 75	Fecha:11/03/2008			
3	LUGAR	FECHA	HORA		Min
4	10	03/06/2008	12:39:57	Rosario	
5	20	03/06/2008	12:42:34	Miraflores	00:02:37
6	30	03/06/2008	12:54:01	Av Americas	00:11:27
7	40	03/06/2008	13:11:24	UDENAR	00:17:23
8	40	03/06/2008	13:18:03	UDENAR	00:06:39
9	50	03/06/2008	13:30:02	I. Panaderia	00:11:59
10	30	03/06/2008	13:36:46	Av Americas	00:06:44
11	20	03/06/2008	13:51:26	Miraflores	00:14:40
12	10	03/06/2008	13:55:53	Rosario	00:04:27
13	10	03/06/2008	14:40:19	Rosario	
14	20	03/06/2008	14:43:38	Miraflores	00:03:19
15	30	03/06/2008	14:54:23	Av Americas	00:10:45
16	40	03/06/2008	15:16:18	UDENAR	00:21:55
17	40	03/06/2008	15:19:14	UDENAR	00:02:56
18	50	03/06/2008	15:30:48	I. Panaderia	00:11:34
19	30	03/06/2008	15:37:00	Av Americas	00:06:12
20	20	03/06/2008	15:51:33	Miraflores	00:14:33
21	10	03/06/2008	15:56:27	Rosario	00:04:54
22	10	03/06/2008	15:59:41	Rosario	00:03:14
23	10	03/06/2008	16:57:41	Rosario	
24	20	03/06/2008	17:04:36	Miraflores	00:06:54
25	30	03/06/2008	17:15:52	Av Americas	00:11:17
26	40	03/06/2008	17:32:14	UDENAR	00:16:22
27	40	03/06/2008	17:37:57	UDENAR	00:05:43
28	50	03/06/2008	17:51:32	I. Panaderia	00:13:35
29	30	03/06/2008	17:56:59	Av Americas	00:05:27
30	20	03/06/2008	18:13:17	Miraflores	00:16:18

En la tarjeta de control manual de tiempos que maneja la empresa se marca con recuadro de color los tiempos registrados por el reloj mecánico, al igual se resalta con colores los datos obtenidos por el módulo móvil del control automático, en los mismos sitios. Se demuestra que el sistema desarrollado cumple con su propósito.

#### 4. CONCLUSIONES

El desarrollo de trabajo de grado cumple con los objetivos propuestos al inicio; como resultado se obtiene un sistema prototipo de control de tiempo de buses automático, confiable y eficiente que cumple con todos los requerimientos para el desempeño de esta tarea.

La comunicación inalámbrica implementada en los módulos fijo y móvil, permite realizar el control de tiempo de manera automática evitando así el estacionamiento del automotor, ya que el área de influencia del transmisor tiene un radio de 50 metros aproximadamente (según las pruebas experimentales realizadas) y el receptor no presenta ningún inconveniente en la recepción de datos, al pasar por el área de influencia del transmisor a una velocidad de 60Km/h (velocidad máxima permitida en el casco urbano). Por lo anterior se puede afirmar que al implementar el sistema en todo el medio de transporte de buses urbanos disminuirá la congestión de tráfico vehicular en los puntos de control y por ende bajaría el índice de accidentes de tránsito en la ciudad, previniendo traumas, tanto físicos como psicológicos para los transeúntes y usuarios vehiculares.

Gracias a los avances tecnológicos y a la miniaturización de los componentes electrónicos se consigue, hacer que los tres módulos desarrollados en el proyecto sean equipos portátiles, lo que permite su fácil instalación y ubicación, son sencillos de utilizar y están contruidos pensando en la comodidad del usuario y en la buena presentación de los dispositivos.

Al implementar a cada uno de los dispositivos una etapa de alimentación de voltaje ininterrumpida (UPS) que permite la conmutación automática de la fuente principal (red eléctrica o batería del automotor), a fuente de respaldo (Acumuladores), se garantiza el funcionamiento del sistema por varias horas en caso de presentarse una falla en la conexión de la fuente principal.

Con el desarrollo de la interfaz USB, se ha suprimido de los computadores personales los puertos de comunicación serial y paralelo debido a que el estándar USB provee una interfaz única, practica y fácil de usar para la gran mayoría de periféricos, además permite la conexión simultanea de un gran número de dispositivos. Se implementa este tipo de comunicación en el presente proyecto, por ser compatible con las nuevas tecnologías.

A pesar que el departamento de Nariño no posee grandes industrias, se puede aplicar los conocimientos adquiridos en la carrera de ingeniería electrónica para mejorar e innovar algunas tareas de las empresas de nuestra región contribuyendo con el progreso y desarrollo tecnológico de la misma.

## **5. RECOMENDACIONES**

- Implementar una comunicación inalámbrica full duplex, para realizar la configuración de los módulos fijos, esto se puede hacer con la utilización de dispositivos transceiver RF.
- Desarrollar un software de aplicación con mejor funcionalidad que además de almacenar y visualizar los datos permita el debido procesamiento y análisis de los mismos según el requerimiento de la empresa.
- Adicionar un modulo para realizar la supervisión remota en los puntos de despacho de cada ruta.

## BIBLIOGRAFIA

DS1307 Serial Real -Time Clock datasheet [Online]. Dallas Semiconductor. 2000. Disponible en Internet: < [http://www.maxim-ic.com/quick\\_view2.cfm/qv\\_pk/2688](http://www.maxim-ic.com/quick_view2.cfm/qv_pk/2688)>.

DUQUE, Edison. Curso Básico De Microcontroladores PIC. Colombia: CEKIT S.A. 1997. 115p.

EEPROM I2C 24LC256 datasheet [Online]. Microchip Technology Inc. 2004. Disponible en Internet: <[http://www.microchip.com/stellent/idcplg?IdcService=SS\\_GET\\_PAGE&nodeId=2057&ty=&dt=Data+Sheets&section=Data+Sheets&ssUserText=24LC256](http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2057&ty=&dt=Data+Sheets&section=Data+Sheets&ssUserText=24LC256)>.

I<sup>2</sup>C-Bus Specification And User Manual [Online]. Philips Semiconductors. Versión 3.0, junio 19, 2007. Disponible en Internet: < <http://www.semiconductors.philips.com/i2c>>; 15p.

INSTITUTO COLOMBIANO DE NORMAS TECNICAS Y CERTIFICACION (ICONTEC). Documentación. Presentación de Tesis, trabajos de grado y otros trabajos de investigación. Quinta actualización. Santa Fé de Bogotá: ICONTEC. 112 p.: il. (NTC 1486).

LOPEZ, Eric. Tutorial Del Protocolo RS-232. [Online]. España: Ingeniería en Microcontroladores, enero del 2007. Disponible en Internet: <[wk3.iespana.es/Serial\\_RS232.pdf](http://wk3.iespana.es/Serial_RS232.pdf)>; 11p.

MALVINO, Albert Paul. Principios De Electrónica. 6 ed. España: McGraw Hill, 2000, 1128p.

Ortiz, Hernán. Libro Electrónico Para La Asignatura De Telemática [CD ROM] [Pasto Nariño]: UDENAR, 2004. Medios de transmisión.

PALACIOS, Enrique, DOMINGUEZ, Fernando, LOPEZ, Lucas. Microcontrolador PIC16F84 Desarrollo de Proyectos. México: Alfaomega. 2004. 648p.

PIC16F628A datasheet [Online]. Microchip Technology Inc. 2005. Disponible en Internet: < [http://www.microchip.com/stellent/idcplg?IdcService=SS\\_GET\\_PAGE&nodeId=1335&d DocName=en010210](http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1335&d DocName=en010210)>.

PIC18F2550 datasheet [Online]. Microchip Technology Inc. 2004. Disponible en Internet:< [http://www.microchip.com/stellent/idcplg?IdcService=SS\\_GET\\_PAGE&nodeId=1335&d DocName=en010280](http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1335&d DocName=en010280)>.

RLP434A datasheet [Online]. Laipac Technology Inc. 2001. Disponible en Internet: < [http://www.laipac.com/easy\\_434\\_eng.htm](http://www.laipac.com/easy_434_eng.htm) >.

STALLINGS, William. Comunicaciones Y Redes De Computadores. 6 ed. España: Prentice Hall, 2000, 748p.

TLP434A datasheet [Online]. Laipac Technology Inc. 2001. Disponible en Internet: < [http://www.laipac.com/easy\\_434\\_eng.htm](http://www.laipac.com/easy_434_eng.htm) >.

TOMASI, Wayne. Sistema De Comunicaciones. 4 ed. México: Prentice Hall, 2003, 976p.

Universal Serial Bus Revision 2.0 specification [Online]. [www.usb.org](http://www.usb.org). Revisión 2.0 Abril 27, 2000. Disponible en Internet: [http://www.usb.org/developers/docs/usb\\_20.pdf](http://www.usb.org/developers/docs/usb_20.pdf); 650p.

WLP434 RF & Cordless Phone Antenna [Online]. Laipac Technology Inc. 2001. Disponible en Internet: < [http://www.laipac.com/ant\\_WLP\\_434\\_eng.htm](http://www.laipac.com/ant_WLP_434_eng.htm)>.

# **ANEXOS**

**ANEXO A. Especificaciones técnicas de dispositivos. (a) módulo fijo, (b) módulo de configuración y (c) módulo móvil.**

<b>Módulo Fijo</b>	
Alimentación	120 VAC 60 Hz
Voltaje de salida	12 VDC, 5VDC
Batería de Respaldo	4,8 VDC, 2700mAh
Duración de Baterías	90 horas
Consumo de corriente en operación normal	27mA
Consumo de corriente en carga de baterías	183mA
Modulación	ASK
Frecuencia de sintonización	433,92Mhz
Alcance promedio	50 mts
Voltaje de operación del transmisor	3 VDC
Potencia del transmisor	10 mW
Resolución reloj(RTC)	Segundos
Puerto de comunicación	Serial, 2.4Kbps, conector DB-9
Temperatura promedio	30 °C
Peso	550 gramos
Dimensiones	103mm x 69mm x 48mm

**(a)**

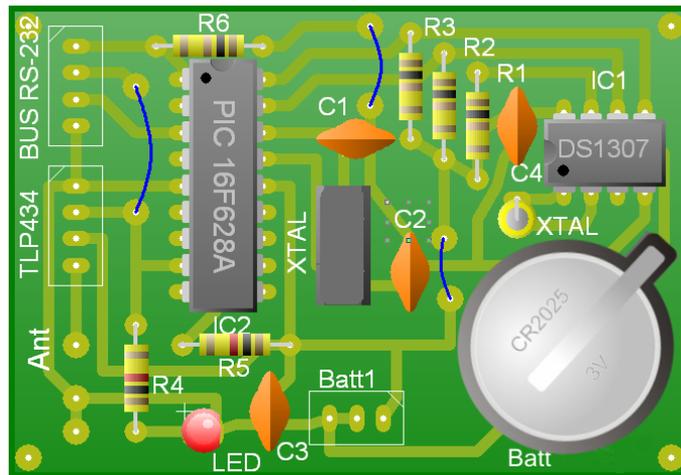
<b>Módulo Configuración</b>	
Alimentación	4,8 VDC, 2700mAh
Duración de baterías	130 horas
Consumo de corriente	18mA
Interfaz	HMI
Pantalla LCD	2x16 caracteres
Teclado	Alfanumérico, 12 caracteres
Puerto de comunicación	Serial, 2.4Kbps, conector DB-9
Resolución reloj (RTC)	Segundos
Temperatura promedio	30 °C
Peso	450 gramos
Dimensiones	55mm x 155mm x 97mm

**(b)**

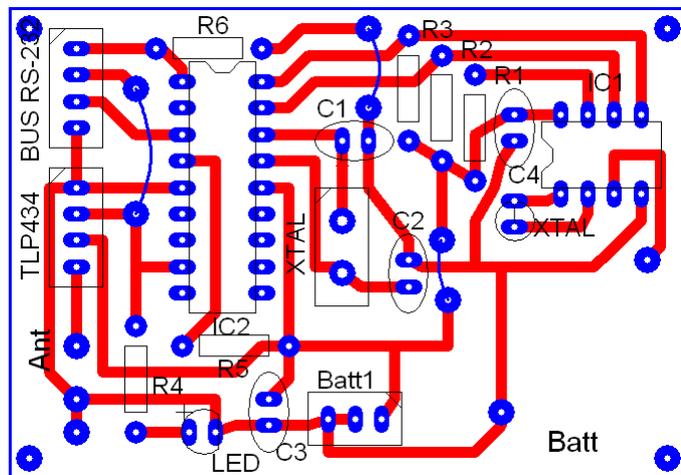
<b>Módulo Móvil</b>	
Alimentación	12 VDC
Voltaje de salida	5VDC
Batería de respaldo	4,8 VDC, 2700mAh
Duración de baterías	85 horas
Consumo de corriente en operación normal	31mA
Consumo de corriente en carga de baterías	180mA
Demodulación	ASK
Frecuencia de sintonización	433,92 Mhz
Antena	WLP434
Voltaje de operación del receptor	5 VDC
Capacidad de memoria	32 Kbytes
Puerto de comunicación	Serial, USB (CDC), 9.6Kbps, conector USB tipo B
Pantalla LCD	2x16 caracteres
Temperatura promedio	30 °C
Peso	400 gramos
Dimensiones	90mm x 109mm x 69mm

(c)

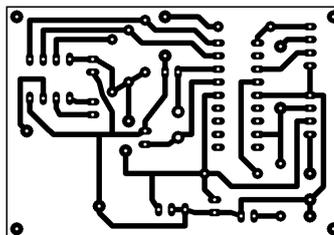
**ANEXO B. Placa de circuito impreso módulo fijo. (a) Vista componentes 3D, (b) cara de componentes y (c) cara de pistas.**



(a)

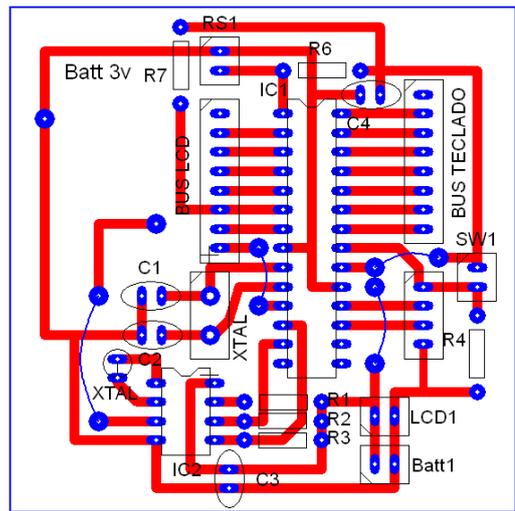
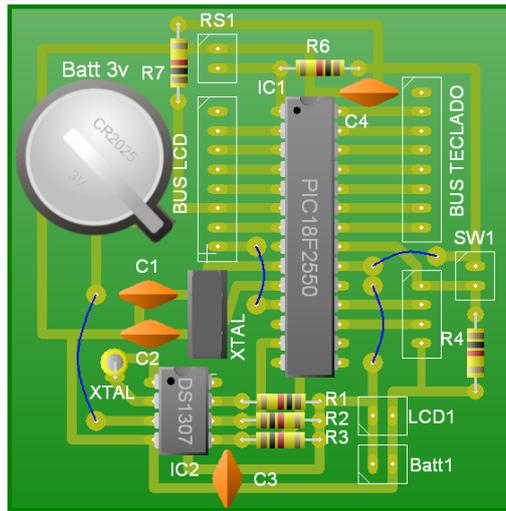


(b)

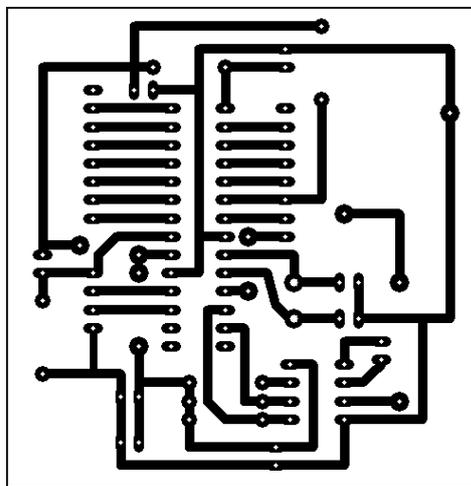


(c)

**ANEXO C. Placa de circuito impreso módulo de configuración. (a) Vista componentes 3D, (b) cara de componentes y (c) cara de pistas.**

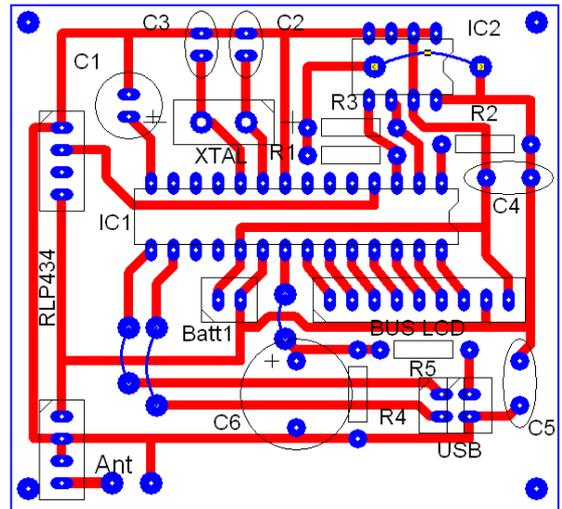
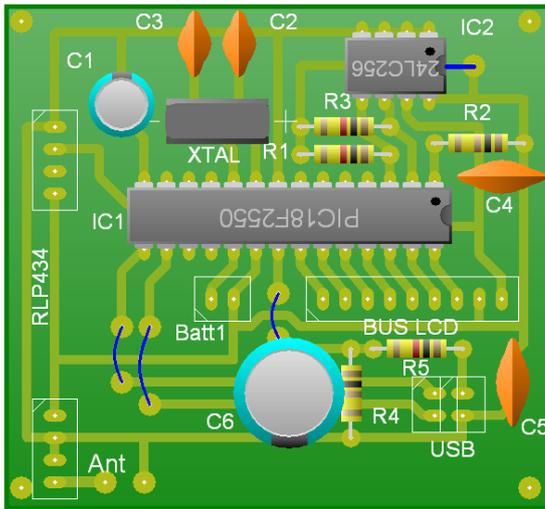


**(a) (b)**

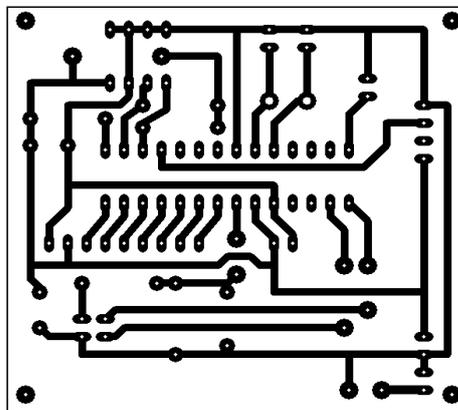


**(c)**

ANEXO D. Placa de circuito impreso módulo móvil. (a) Vista componentes 3D, (b) cara de componentes y (c) cara de pistas.

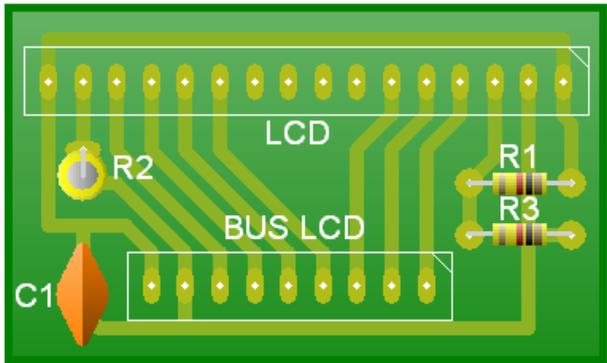


(a) (b)

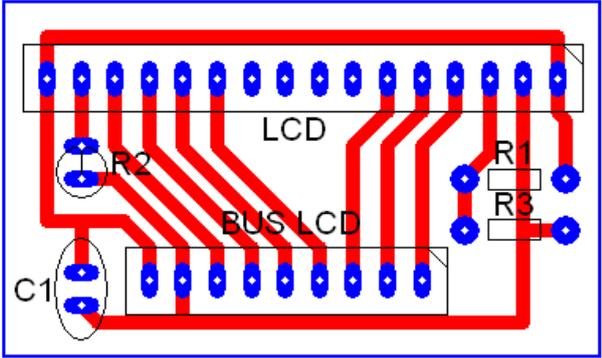


(c)

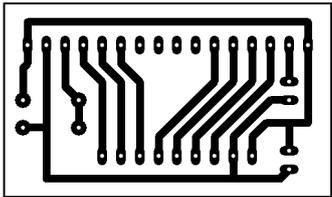
**ANEXO E. Placa de circuito impreso pantalla LCD. (a) Vista componentes 3D, (b) cara de componentes y (c) cara de pistas.**



(a)

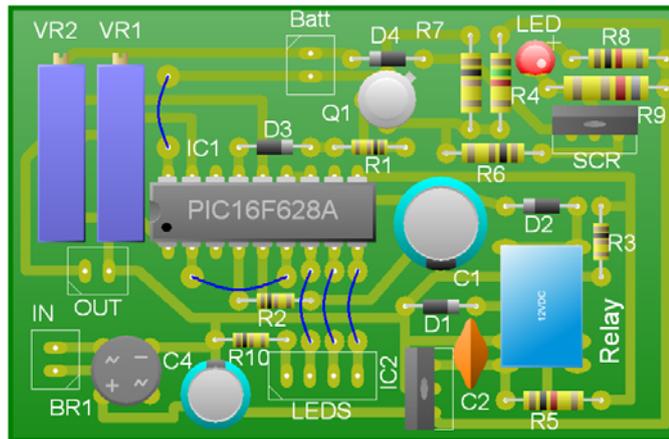


(b)

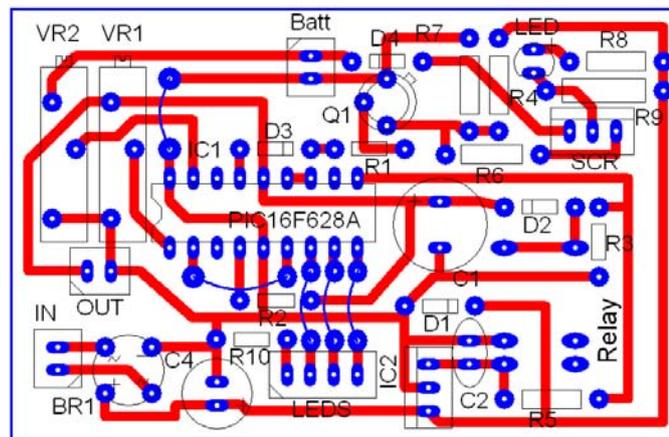


(c)

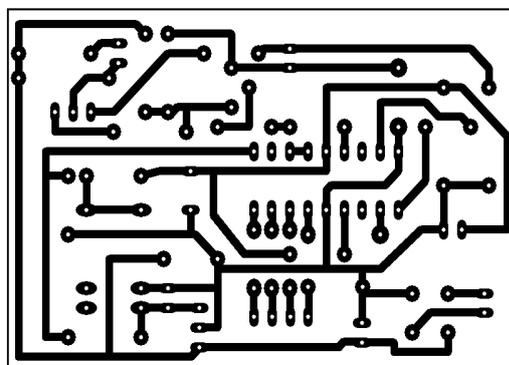
**ANEXO F. Placa de circuito impreso UPS módulo fijo. (a) Vista componentes 3D, (b) cara de componentes y (c) cara de pistas.**



(a)

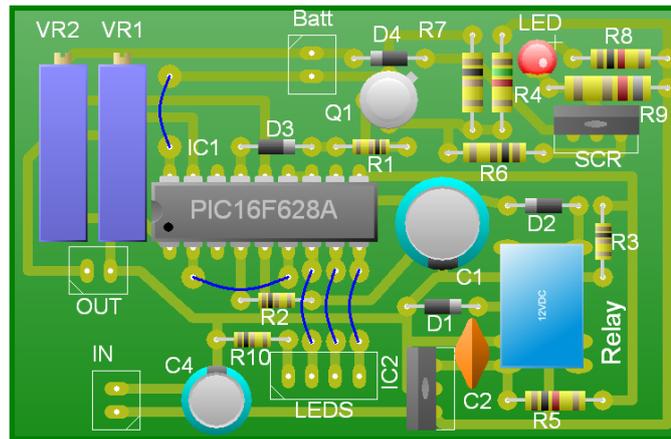


(b)

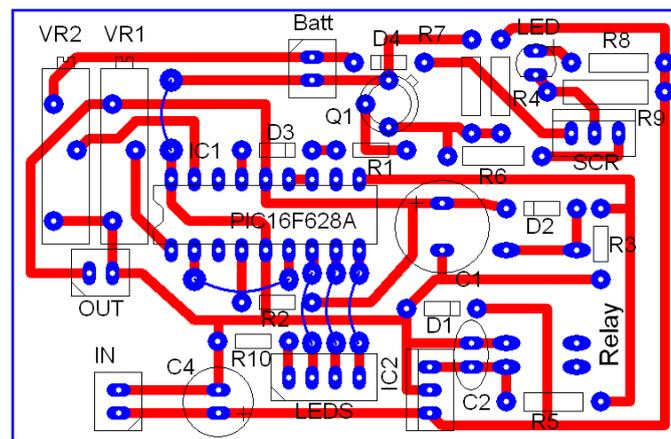


(c)

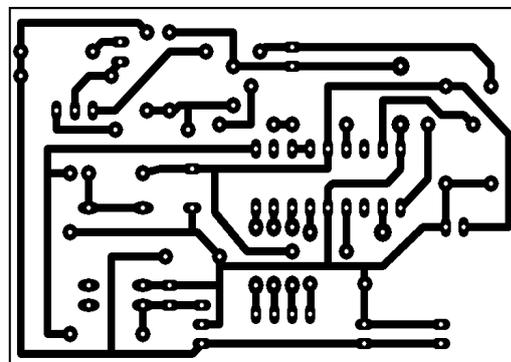
**ANEXO G. Placa de circuito impreso UPS módulo móvil. (a) Vista componentes 3D, (b) cara de componentes y (c) cara de pistas.**



**(a)**

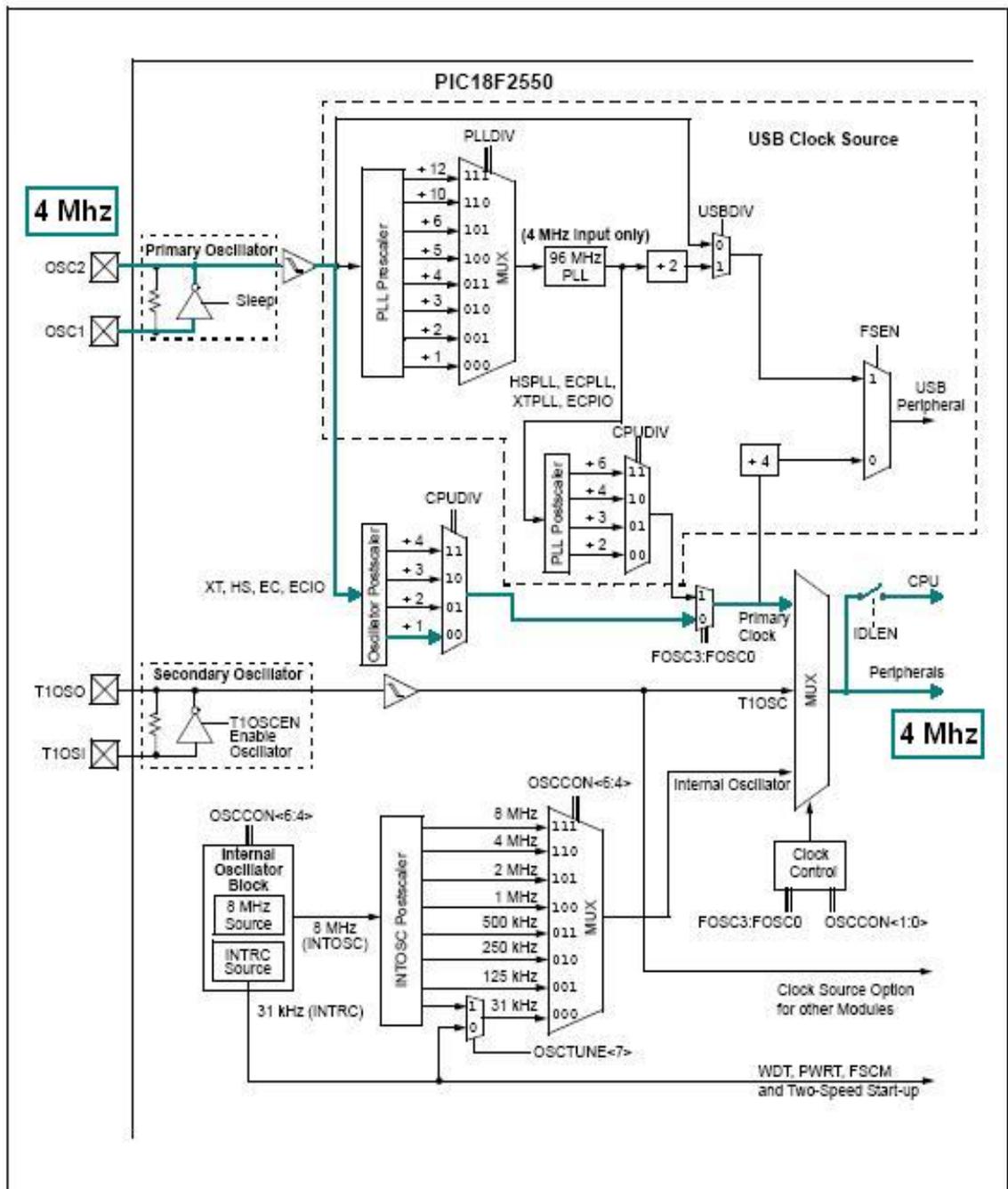


**(b)**



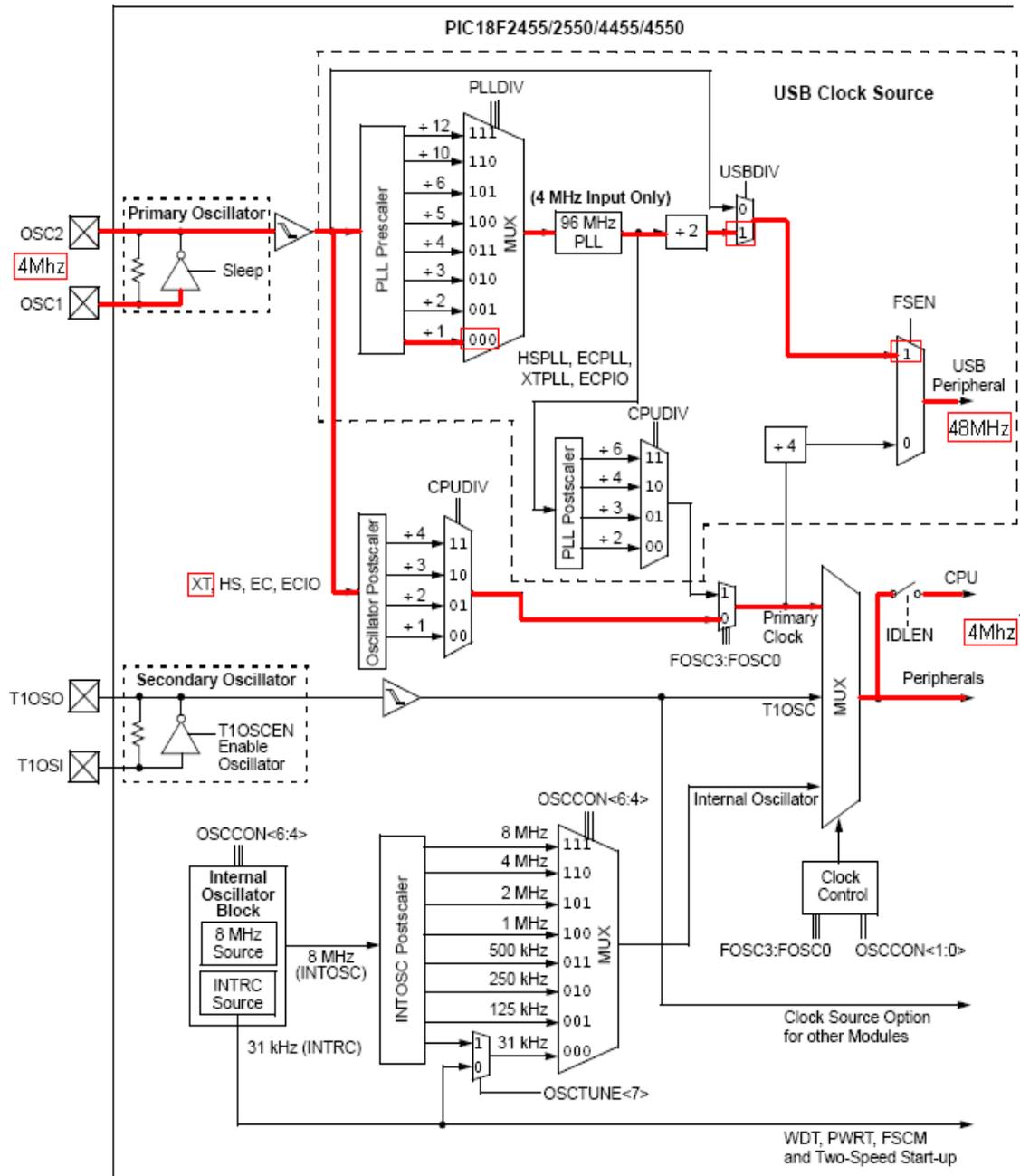
**(c)**

## ANEXO H. Diagrama del módulo oscilador dispositivo de configuración



Fuente: PIC16F628A datasheet [Online]. Microchip Technology Inc.

## ANEXO I. Diagrama del módulo oscilador dispositivo móvil



Fuente: PIC18F2550 datasheet [Online]. Microchip Technology Inc.

**ANEXO J. Mapa de la ciudad San Juan de Pasto, en el que se identifican la ubicación de los puntos de control y el recorrido de la ruta N° 12 de Autobuses del sur.**

