



PONTIFICIA **UNIVERSIDAD CATÓLICA** DEL PERÚ

Esta obra ha sido publicada bajo la licencia Creative Commons
Reconocimiento-No comercial-Compartir bajo la misma licencia 2.5 Perú.

Para ver una copia de dicha licencia, visite
<http://creativecommons.org/licenses/by-nc-sa/2.5/pe/>



PONTIFICIA UNIVERSIDAD CATOLICA DEL PERU

FACULTAD DE CIENCIAS E INGENIERIA



**IMPLEMENTACION DE UN INTERFAZ PARA
TRANSMISION DE DATOS AIRE TIERRA EN
AEROPUERTOS EN VHF AM**

Tesis para optar el titulo de Ingeniero Electrónico

Ricardo Cárdenas Abanto

Pando – Lima

2006

RESUMEN

El presente documento tiene como finalidad documentar el desarrollo y las pruebas realizadas del prototipo de interfase que se describe a continuación. En la actualidad los sistemas de comunicaciones aeronáuticas tienen la tendencia a ser automatizadas, es decir, que datos como la posición, altitud y otros puedan ser enviados desde las aeronaves hacia el centro de control de manera automática sin que el piloto tenga que leer sus instrumentos de navegación y reportárselo al centro de control. Existen equipos que pueden realizar tal tarea que son una combinación de un receptor GPS y un transmisor de radiofrecuencia con modulación digital. El costo de dichos equipos son elevados para algunos casos, como para las empresas aéreas que operan en el aeródromo de Nasca. La aplicación desarrollada en este proyecto trata de suplir esta necesidad utilizando sistemas relativamente baratos como es el caso de transmisores VHF AM analógicos comunes que todas las aeronaves poseen.

Estos equipos y su ancho de banda fueron el punto de partida para desarrollar una interfase que pueda capturar datos de latitud y longitud usando el sistema GPS y pueda a su vez transmitirlos a una estación base usando las comunes radios analógicas VHF AM. La única manera de hacer posible este esquema es enviar tonos que estén en el rango de audio. Se utilizaron generadores y detectores de tonos duales multifrecuencia DTMF para modular los transmisores VHF AM y a su vez estos puedan ser decodificados. La primera etapa se desarrollo usando una PC

que maneja por el puerto paralelo los circuitos DTMF, resultando satisfactorias las primeras pruebas. La segunda etapa consiste en eliminar la PC y reemplazarla por un microcontrolador de manera que la interfase de a bordo pueda ser totalmente automática, además de acoplar en forma serial un receptor GPS que captura la información de posición, la procesa y la envía en forma de tonos DTMF. En la estación base se reciben los tonos y son demodulados por un receptor VHF AM y luego enviados a otro dispositivo que decodifica los tonos, reprocesa los datos y obtiene la posición del móvil. Estos datos son enviados a otra PC en forma serial para su visualización en un monitor similar a la pantalla de un RADAR.

Este proyecto se desarrollo íntegramente en el Área de Proyectos Especiales de Aeronavegación de CORPAC con la finalidad de proponer en el Perú una solución alternativa de monitoreo automático de posición a las empresas aéreas que no están en condiciones de adquirir costosos equipos con modulación digital.

Agradecimientos

Quiero agradecer a todas las personas que me apoyaron para poder desarrollar este proyecto, desde aquellos que tomaron la decisión de llevarlo a cabo así como a aquellos que en el momento de las pruebas estuvieron cargando las baterías o instalando las antenas.

Personalmente quiero agradecer al Ing. José Luís Paredes por la confianza depositada en mí, así como al Ing. Tomas Sheen Jefe del Proyecto por todo el apoyo técnico y teórico transmitido.

También agradecer al Ing. Percy Gamboa desarrollador del programa de monitoreo con quien trabajamos juntos para el éxito de este proyecto.

También agradecer al Prof. David Chávez por el apoyo y el ánimo para terminar de redactar este documento.

También quiero agradecer a mi familia por el apoyo para poder culminar la elaboración de este documento.

INDICE

1. INTRODUCCION.....	1
2. CARACTERISTICAS DE TRANSMISION EN VHF-AM.....	4
2.1. Espectro Electromagnético.....	6
2.2. Modos de Transmisión.....	8
2.3. Técnicas de Modulación Analógicos.....	8
2.3.1. Modulación de Amplitud.....	9
2.3.2. Modulación de Frecuencia.....	11
2.3.3. Modulación de Fase	12
2.4. Técnicas de Modulación Digital.....	12
2.5. Señales de Audio.....	14
2.5.1 Análisis de Interferencias en una radio VHF AM.....	14
2.5.1.1 Amplificador de Radiofrecuencia (RF AMP).....	15
2.5.1.2 Mezclador (MIXER).....	16
3. SISTEMAS DE POSICIONAMIENTO GEOGRAFICO.....	18
3.1. Sistemas Geodésicos.....	18
3.2. Sistemas de Posicionamiento Global (GPS)	20
3.3. Diferencial GPS	23
3.4. Protocolo NMEA 0183.....	23
4. PROTOCOLOS DE TRANSMISION DE DATOS.....	27
4.1. Características de los protocolos	27
4.2. Protocolos orientados a bits	28
4.3. protocolos orientados a byte	29

4.4. Protocolos Asíncronos.....	30
4.5. Protocolos Síncronos.....	30
5. DISPOSITIVOS ELECTRONICOS.....	31
5.1. Amplificadores Operacionales	31
5.2. Transistores	34
6. MULTICANALIZACION Y ACCESO AL MEDIO	36
6.1. Multicanalización	36
6.1.1. TDMA	36
6.1.2. FDMA	36
6.2. Métodos de Acceso al Medio	37
6.2.1. CSMA/CD	38
6.2.2. CSMA/CA	40
7. DESARROLLO DE LA APLICACIÓN	43
7.1. Estudio de la forma de modulación digital.....	43
7.2. Análisis de las características de los Radios VHF-AM	45
7.3. Diseño del protocolo de transmisión de datos	46
7.3.1. Análisis de datos recibidos del receptor GPS	46
7.3.2. Diseño de la trama	47
7.4. Primera Etapa	50
7.4.1. Diseño del Circuito Electrónico	51
7.4.2. Programa en Turbo C/C++	54
7.4.3. Primera Prueba.- Funcionamiento de protocolo	56
7.4.3.1 Resultados	56

7.4.3.2	Análisis de resultado	56
7.4.3.3	Primera modificación y resultados	56
7.4.3.4	Segunda modificación y resultados	57
7.4.4.	Segunda Prueba.- PTT automático	54
7.4.5.	Tercera Prueba.- Cobertura y factores externos	61
7.4.5.1	Resultados	61
7.4.5.2	Modificaciones y resultados	63
7.4.6.	Análisis de errores del sistema	63
7.4.7.	Cuarta Prueba.- Pruebas de campo en Nasca	64
7.4.7.1	Observaciones.....	64
7.4.7.2	Modificaciones en sitio	65
7.4.8.	Análisis de rendimiento	66
7.5.	Segunda Etapa	67
7.5.1.	Estudio del método de Acceso al medio	67
7.5.1.1.	Caso TDMA sincroniza con reloj transmitido de tierra ...	68
7.5.1.2.	Caso CSMA	68
7.5.1.3	Caso de transmisión de posición por petición	69
7.5.1.4.	Caso TDMA que sincroniza con hora UTC del GPS	69
7.5.1.5.	Opción escogida	70
7.5.2.	Diseño del Circuito Electrónico	71
7.5.2.1.	Estudio del microcontrolador	72
7.5.2.2.	Estudio de Transceiver DTMF 8888	73
7.5.2.3.	Circuito de generación decodificación de tonos DTMF ..	74

7.5.2.4. Circuito de interfase para entradas y salidas	76
7.5.2.5. Circuito de transmisión y recepción serial	77
7.5.2.6. Análisis de parámetros eléctricos	77
7.5.2.7. Diagrama Esquemático	78
7.5.3. Software.....	79
7.5.3.1. Programa en Lenguaje ensamblador	79
7.5.3.2. Algoritmo de retardo	80
7.5.4. Alojamiento Metálico	81
7.5.5. Pruebas y Resultados	82
7.6. Análisis de mejoras futuras al proyecto	84
7.6.1 Equipos GPS	84
7.6.2 Incremento de aeronaves y acceso al medio	87
8. CONCLUSIONES	88
9. OBSERVACIONES Y RECOMENDACIONES.....	90

LISTA DE TABLAS

Tabla 2.1: Asignación de frecuencias

Tabla 7.1: Tabla de frecuencias DTMF

Tabla 7.2: Registros del 8888

Tabla 7.3: Descripción del Registro de Control A del 8888

Tabla 7.4: Descripción del Registro de Control B del 8888

Tabla 7.5: Descripción del registro de estado del 8888

Tabla 7.6: Consumo de corriente de tarjeta

Tabla 7.7: Cuadro comparativo de equipos GPS

LISTA DE FIGURAS

Figura 2.1: Espectro de frecuencia AM y ancho de banda

Figura 2.2: Diagrama de modulación en frecuencia

Figura 2.3: Diagrama de bloques de recepción VHF AM

Figura 2.4: Sumatoria de dos señales

Figura 2.5: Sumatoria de dos señales más interferencia

Figura 3.1: Sistema de coordenadas WGS84

Figura 5.1: Configuración básica de OPAMPS

Figura 5.2: Amplificador Inversor

Figura 5.3: Amplificador No Inversor

Figura 5.4: Amplificador seguidor de voltaje

Figura 5.5: Símbolos de transistores bipolares

Figura 6.1: Diagrama TDMA

Figura 6.2: Diagrama FDMA

Figura 6.3: Diagrama de acceso al medio

Figura 6.4: Diagrama de CSMA/CA

Figura 7.1: Ancho de banda del audio

Figura 7.2: Ancho de banda del canal

Figura 7.3: Coordenadas cartesianas de ubicación

Figura 7.4: Diagrama de tonos DTMF

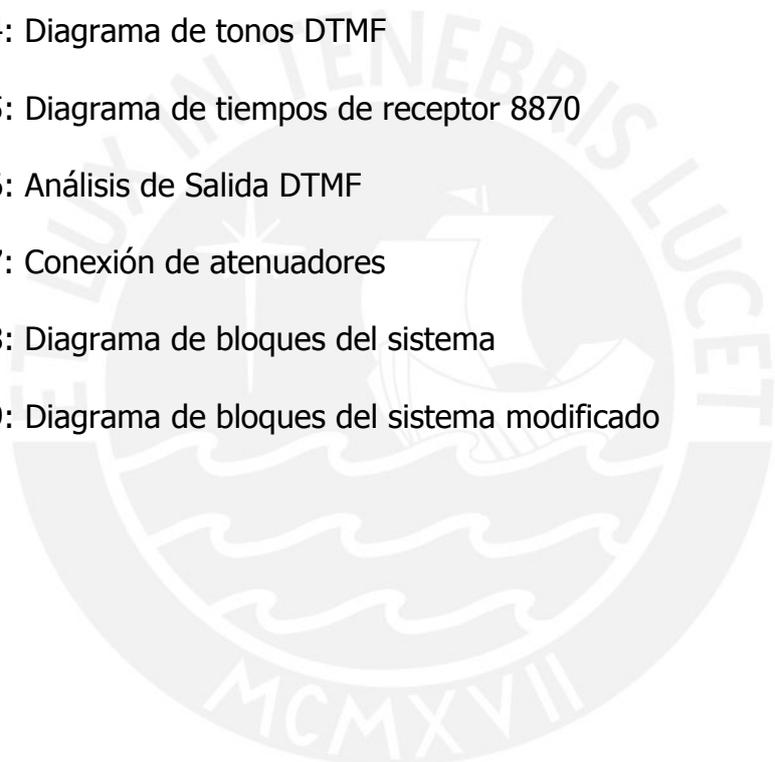
Figura 7.5: Diagrama de tiempos de receptor 8870

Figura 7.6: Análisis de Salida DTMF

Figura 7.7: Conexión de atenuadores

Figura 7.8: Diagrama de bloques del sistema

Figura 7.9: Diagrama de bloques del sistema modificado



1. INTRODUCCION

Como es de conocimiento general, el desarrollo de la tecnología ha traído innumerables cambios en los sistemas de comunicaciones existentes, por consiguiente, en los sistemas de comunicaciones aeronáuticos se han desarrollado muchos sistemas que buscan hacer de la aviación, un medio más eficiente, seguro y confiable.

La **OACI** *Organización de Aviación Civil Internacional*, es una entidad que se encarga de dirigir todos los temas concernientes a la aviación en todo el mundo. Después de algunos años de estudio elaboro un Proyecto de recomendaciones acerca de cómo debe ser el futuro de las comunicaciones, la navegación y la vigilancia aeronáutica.

Con este fin se acordó la implantación a nivel mundial del sistema Comunicaciones Navegación Vigilancia/ Gestión de Trafico Aéreo **CNS/ATM** por sus siglas en ingles, sistema al cual todos los países deberían migrar durante los siguientes años.

En resumen este sistema comprende:

En el área de **Comunicaciones**: Comprende el **ATN** *Aeronautical Telecommunications Network* que es una plataforma digital de comunicaciones aeronáuticas que debe soportar las actuales redes de comunicaciones públicas (Frame Relay, ISDN, ATM, etc.).

En el área de **Navegación**: Comprende sistemas de navegación utilizando los satélites, más específicamente el Sistema Americano **GPS** *Global Positioning System*.

En el área de **Vigilancia**: Comprende el **CPDLC** *Controller Pilot Data Link Communications* y el **ADS** *Automatic Dependent Surveillance*.

El ADS consiste en la vigilancia que se le hace a una determinada aeronave mediante la visualización de su posición en una pantalla similar a un radar, además de cierta información adicional como velocidad, rumbo, etc.

CORPAC a la vanguardia de la transición al CNS/ATM en Sudamérica, empezó a adquirir radios digitales, que trabajan con el denominado **Modo 2** que implica modulación digital estandarizado **D8PSK**. Además en convenio con la compañía norteamericana ARINC se instalaron en Lima, equipos para hacer las pruebas de ADS y CPDLC.

Posteriormente se desarrolló en CORPAC un ADS experimental, utilizando un equipo de la empresa ARINC montada en una aeronave, la que enviaba continuamente su posición (latitud, longitud) recibida de un equipo GPS.

En tierra se recibía esta información con la ayuda de las radios digitales Harris en Modo 2 y en una pantalla se podía visualizar el movimiento del "blanco" usando un software de visualización desarrollado en Visual Basic.

Estas pruebas fueron hechas en los aeropuertos de Lima, Cuzco y Nasca con resultados satisfactorios. Luego de las pruebas en Nasca, que es un aeropuerto de alto tráfico pero solo para avionetas, surge la necesidad de desarrollar un sistema ADS de bajo costo, ya que las empresas aéreas que operan en esa ciudad no estarían en condiciones de adquirir dichos equipos por su elevado costo.

Es en ese momento que se decide desarrollar experimentalmente un sistema que usando radios convencionales con modulación en amplitud que las avionetas poseen pudiera transmitir información de su posición captada de un equipo GPS.

En general este prototipo debería tener la capacidad de capturar información de un equipo receptor GPS mediante una interfase serial, luego esta información debería procesarse de tal forma que pudiera modular ondas analógicas de radios VHF con modulación en amplitud. Debería ser capaz también en el extremo remoto de decodificar la información enviada.

Bajo este contexto y surgida esta necesidad se empieza el desarrollo de este proyecto cuyos prototipos se encuentran finalizados y realizadas las pruebas correspondientes en el aeropuerto de Nasca.

En este proyecto se desarrollaron programas en Lenguaje Turbo C/C++ y en Lenguaje Ensamblador para el microcontrolador PIC16F877, además se fabricaron tarjetas electrónicas como interfaces entre los transmisores VHF y los receptores GPS.

2. CARACTERÍSTICAS DE TRANSMISION EN VHF-AM

CORPAC Corporación Peruana de Aeropuertos y Aviación Comercial es una entidad del Estado peruano cuya función es la de velar y otorgar las facilidades operacionales necesarias para que la aeronavegación se desarrolle sin contratiempos a lo largo de todo el territorio peruano.

Estos servicios son los Servicios de Aeronavegación y comprenden el Servicio de Transito Aéreo, Meteorología, Información Aeronáutica y Alerta, estos servicios están soportados por los Sistemas de Comunicaciones, Navegación, Vigilancia, Ayudas Luminosas y Energía. El objetivo final de estos servicios es el de brindar seguridad a las operaciones aéreas y a los pasajeros.

Aeropuertos: Son las áreas físicas donde se realizan las maniobras de aterrizaje y despegue de aeronaves.

Gestión del Transito Aéreo: La gestión del transito aéreo debe posibilitar que las operaciones de aeronaves cumplan con sus tiempos planificados en la partida como en la llegada y con sus trayectorias de vuelo elegidas, con un mínimo de problemas y sin comprometer los niveles de seguridad acordado.

Navegación: Se refiere a los procedimientos que se realizan durante el vuelo en ruta de una aeronave. Se refiere a las rutas aéreas establecidas y a la vigilancia que se hace de esta aeronave.

Comunicaciones: Suministra el intercambio de datos y mensajes aeronáuticos entre los usuarios aeronáuticos y los equipos automáticos.

Una de las plataformas principales es el Sistema de Telecomunicaciones instalado en todo el Perú. Este incluye el Servicio de Comunicaciones Fijas y el Servicio de Comunicaciones Móvil aeronáutico

Comunicaciones Fijas aeronáuticas

En el servicio fijo se utilizarán comunicaciones orales (referidas a la voz) directas y/o técnicas de intercambio en las comunicaciones tierra – tierra para fines de los servicios de tránsito aéreo. Se usan circuitos dedicados para la comunicación entre Centros de Control de los distintos aeropuertos del Perú.

También se incluyen los distintos circuitos dedicados internacionales para comunicaciones con Centros de Control de los países limítrofes.

Comunicaciones Móviles aeronáuticas

Incluye la infraestructura y el soporte necesario por parte de Corpac para que un piloto de una aeronave cualquiera establezca comunicación con el controlador de tránsito aéreo dispuesto en la torre o en el centro de control de un determinado aeropuerto.

El intercambio de información se hace a través de voz, desde la petición de autorización de despegue, de aterrizaje y la posición actual: latitud y longitud cada cierto punto de chequeo de una ruta determinada.

CORPAC utiliza la banda de frecuencia VHF (*Very High Frequency*) 118 -136 MHz la cual está asignada por el Ministerio de Transportes y Comunicaciones para uso aeronáutico.

Esta información es complementada por la información de un RADAR, solamente en áreas donde se tenga cobertura de este, que es básicamente Lima con un radio de 20 millas de cobertura. Los demás aeropuertos del Perú se limitan a intercambiar información a través de voz utilizando equipos de radiocomunicación en la banda de VHF con modulación en amplitud.

2.1 Espectro Electromagnético

El espectro de frecuencias electromagnéticas está distribuido en un rango de frecuencias casi infinito, se extiende desde las frecuencias *subsónicas* hasta los *rayos cósmicos*. Este espectro total se divide en bandas y cada una tiene sus propias características y son diferentes a las demás. Por ejemplo las características de propagación de las frecuencias del rango de audio son diferentes a las del rango de microondas.

Sabemos que la longitud de onda (λ) es inversamente proporcional a la frecuencia (f) y directamente proporcional a la velocidad de propagación (c).

La relación matemática es como sigue:

$$\text{Longitud de onda} = \text{velocidad} / \text{frecuencia}$$

El espectro total está dividido en sub sectores o bandas, cada una de ellas tiene su nombre y está limitado en frecuencia. A esto se denomina asignación de frecuencias. Por ejemplo la banda de frecuencias asignada para el uso aeronáutico es de 118 a 136 MHz y este rango está comprendido dentro del rango de VHF (*Very High Frequency*).

Según el ITU-R Unión Internacional de Telecomunicaciones - Sección Radiocomunicación, se asignan las frecuencias segunda tabla 2.1 que se presenta a continuación:

TABLA 2.1 Asignación de frecuencias

Numero de banda	Rango de frecuencias	Designaciones
2	30-300 Hz	ELF (frecuencias extremadamente bajas)
3	0.3-3 KHz	VF (frecuencias de voz)
4	3-30 KHz	VLF (frecuencias muy bajas)
5	30-300 KHz	LF (frecuencias bajas)
6	0.3-3 MHz	MF (frecuencias medias)
7	3-30 MHz	HF (frecuencias altas)
8	30-300 MHz	VHF (frecuencias muy altas)
9	0.3-3 GHz	UHF (frecuencias ultra altas)
10	3-30 GHz	SHF (frecuencias súper altas)
11	30-300 GHz	EHF (frecuencias extremadamente altas)
12	0.3-3 THz	Luz Infrarroja
13	3-30 THz	Luz Infrarroja
14	30-300 THz	Luz Infrarroja
15	0.3-3 PHz	Luz visible
16	3-30 PHz	Luz ultravioleta
17	30-300 PHz	Rayos-X
18	0.3-3 EHz	Rayos Gamma
19	3-30 EHz	Rayos cósmicos

Según el Plan Nacional de Atribución de Frecuencias del Ministerio de Transportes y Comunicaciones del Perú se establece que entre 117.75 -136 MHz esta reservada para Comunicaciones Móviles Aeronáuticas.

2.2 Modos de Transmisión

Se denominan modos de transmisión a la forma como los sistemas de comunicaciones manejan el intercambio de información: en una dirección, en dos direcciones al mismo tiempo o solo uno a la vez.

Simplex (SX): Las transmisiones solo pueden ocurrir en una dirección, es decir solo pueden recibir o solo transmitir. Como ejemplo práctico tenemos los sistemas de radiodifusión de radio FM.

Half-Duplex (HDX): En este modo el intercambio de información puede ocurrir en ambos sentidos pero no al mismo tiempo sino que lo hacen uno a la vez. Tenemos como ejemplo los sistemas de radio de dos vías troncalizado en que se usan el sistema de PTT (presionar para hablar).

Full-Duplex (FDX): Aquí las transmisiones pueden ocurrir en ambos sentidos y al mismo tiempo. En este modo una estación puede estar transmitiendo y recibiendo simultáneamente siempre y cuando la que está transmitiendo sea la que está recibiendo.

Full/Full-Duplex (F/FDX): En este modo se puede transmitir y recibir al mismo tiempo sin necesidad de que sea entre las mismas estaciones. La aplicación de este tipo de sistema de comunicaciones son los circuitos de comunicación de datos.

2.3 Técnicas de Modulación Analógicas

Las señales de información deben ser transportadas entre un transmisor y un receptor sobre alguna forma de medio de transmisión. Sin embargo pocas

veces las señales de información se encuentran en la forma adecuada para la transmisión. La modulación la definimos como el proceso de transformar información de su forma original a una forma más adecuada para su transmisión.

Demodulación es el proceso inverso (es decir la onda modulada se convierte a su forma original). La modulación se realiza en el transmisor en un circuito que se llama modulador y la demodulación en el receptor en un circuito que se llama demodulador.

2.3.1 Modulación de Amplitud (AM)

La modulación de amplitud es un proceso mediante el cual se imprime la amplitud de una señal modulante sobre una portadora de frecuencia mucho más alta en forma de cambios de amplitud, la cual se puede irradiar a través de una antena y pueda propagarse a través del espacio libre. A estas frecuencias suficientemente altas que son capaces de irradiarse de manera eficiente por una antena se llaman Radiofrecuencia o simplemente RF.

La modulación en amplitud es relativamente una forma barata de modulación y de baja calidad que se usa básicamente para radiodifusión de señales de audio y video.

Un modulador AM es un aparato no lineal con dos señales de entrada de información: una señal portadora de amplitud constante y alta frecuencia o moduladora y la señal de información o modulante. La resultante se denomina onda modulada en amplitud. Ver figura 2.1.

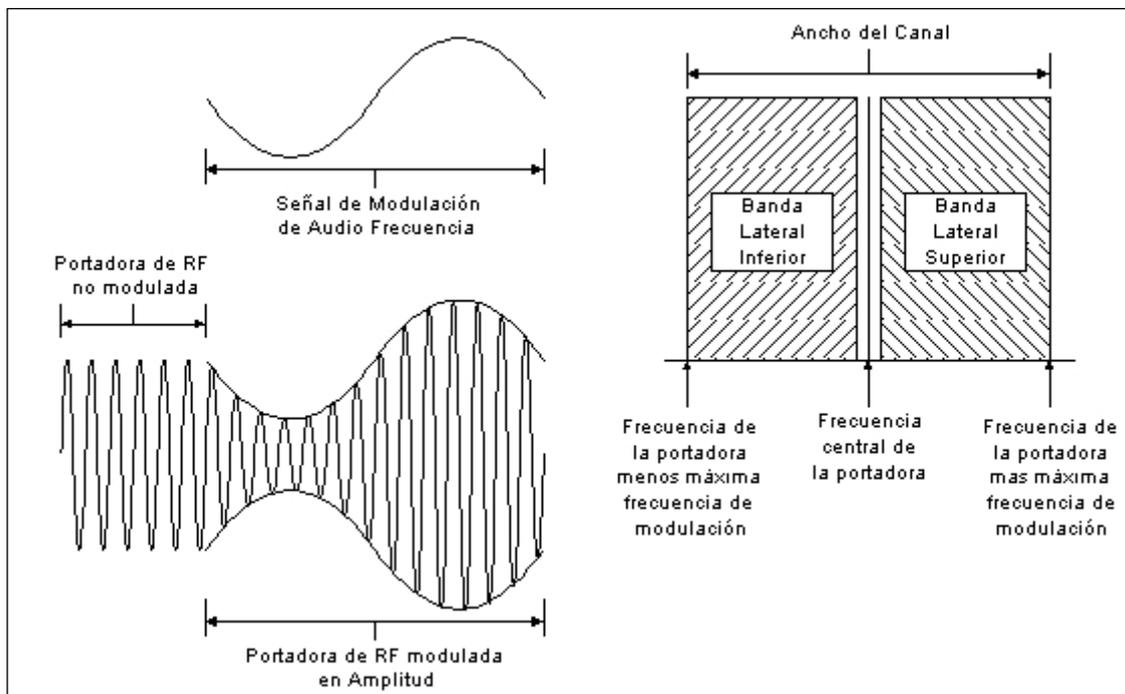


Figura 2.1: Espectro de frecuencia AM y ancho de banda

AM es una mezcla de señales de audiofrecuencia (AF) y radiofrecuencia (RF), de manera que las variaciones de amplitud de la señal de AF (modulación) alteran la amplitud de la señal de RF (portadora). La onda modulada de RF tiene la forma simétrica arriba de la figura 2.1 y debajo de la línea de referencia cero. Las frecuencias laterales superior e inferior iguales a la suma y diferencia de las dos frecuencias originales se generan en el proceso de modulación y aparecen en el espectro de frecuencias inmediatamente arriba y debajo de la frecuencia de la portadora. Por ejemplo, si la frecuencia de la portadora es de 1MHz y la frecuencia de modulación es de 1kHz, se producen arriba y debajo de la frecuencia de portadora las frecuencias laterales superior e inferior de 1.001Mhz y 999kHz respectivamente. Si la señal de modulación varía en frecuencia, crea una banda de frecuencias a

cada lado de la portadora, conocida como la banda lateral superior y banda lateral inferior. El ancho de banda total de la señal modulada queda determinado por la frecuencia de modulación más alta y es igual a la suma de las dos bandas laterales. El ancho de canal total de la señal portadora es igual a la frecuencia de portadora más y menos las dos bandas laterales.

2.3.2 Modulación de Frecuencia

La modulación en frecuencia (FM) es el proceso de combinar una señal de AF (Audio Frecuencia) con otra de RF (Radio Frecuencia), tal que la amplitud de la AF varíe la frecuencia de la RF. Ver figura 2.2.

Si la señal de modulación varía en frecuencia, no tiene efecto en las excursiones máxima y mínima de la frecuencia de portadora, sino que solo determina la rapidez o lentitud con que ocurren las variaciones en la frecuencia.

Es decir, que una frecuencia mas baja de modulación provoca que ocurran variaciones a una tasa más lenta, y una frecuencia mas alta de modulación hace que ocurran a una tasa más rápida. Sin embargo, las variaciones en amplitud de la señal de modulación si afectan las excursiones máxima y mínima de la frecuencia portadora. Una señal de mayor amplitud provoca un mayor cambio en la frecuencia y una señal más pequeña provoca un cambio menor en la frecuencia.

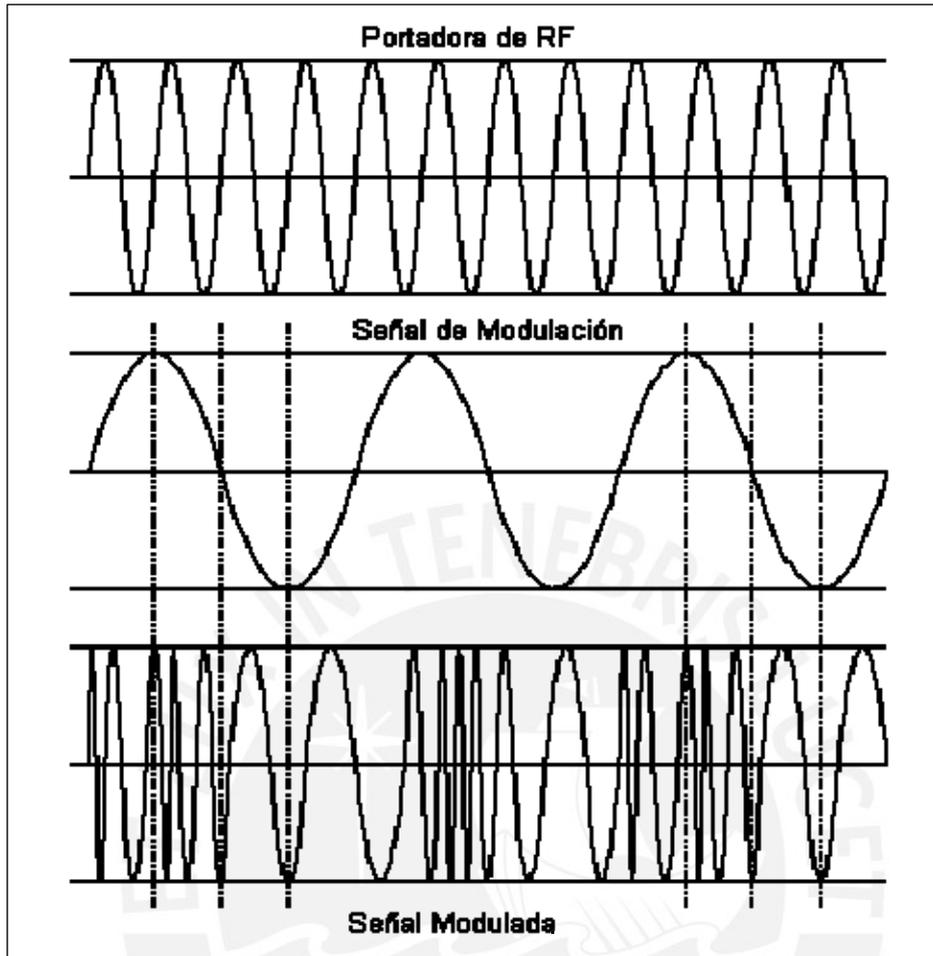


Figura 2.2: Diagrama de Modulación de Frecuencia

2.3.3 Modulación de Fase

Sistema de modulación en el cual la fase de la señal portadora varía o es modulada conforme al valor instantáneo de la amplitud de la señal moduladora.

2.4 Técnicas de Modulación Digital

Modulación de Frecuencia (FSK, Frequency Shift Keying): es un tipo de modulación de frecuencia en la cual la señal modulante desplaza la frecuencia de la onda portadora entre valores discretos predeterminados.

Generalmente la frecuencia instantánea se desplaza entre dos valores discretos denominados frecuencia marca y frecuencia espacio. Este es el método no

coherente de FSK. Existe también el método coherente de FSK en el cual no hay discontinuidades de fase en la onda portadora.

Modulación de Amplitud (ASK, Amplitud Shift Keying): es una forma de modulación en la cual se representan los datos digitales como variaciones de amplitud de la onda portadora.

La forma más simple y común de ASK funciona como un interruptor que apaga/enciende la portadora, de tal forma que la presencia de portadora indica un 1 binario y su ausencia un 0. Este tipo de modulación por desplazamiento on-off es el utilizado para la transmisión de código Morse por radiofrecuencia, siendo conocido el método como operación en onda continua.

Modulación de Fase (PSK, Phase Shift Keying): se codifican los valores binarios como cambios de fase de la señal portadora.

Modulación Diferencial de Fase (DPSK, Differential Phase Shift Keying): consiste en una variación de PSK donde se toma el ángulo de fase del intervalo anterior como referencia para medir la fase de cualquier intervalo de señal.

Modulación de Amplitud de Cuadratura (QAM, Quadrature Amplitude Modulation): Consiste en una combinación de PSK y ASK, es decir, se van a combinar las variaciones de amplitud en referencia al momento de fase en que ocurren con lo cual vamos a poder incluir más bits en los mismos hercios.

Modulación con tonos DTMF, en este caso se tienen tonos duales (dos frecuencias) que están en el rango de las señales de audio.

2.5 Señales de Audio

Las señales de audio se encuentran en el rango de 300 – 3300 Hercios.

2.5.1 Análisis de Interferencias en una Radio VHF-AM

En un sistema de comunicaciones VHF-AM ocurren ciertos tipos de perturbaciones de las comunicaciones debido a interferencias propias del medio de propagación, de la frecuencia de trabajo y factores intrínsecos relacionados con los transmisores y receptores.

A continuación hacemos un análisis de los factores propios de los equipos para lo cual se muestra el diagrama de bloques de un receptor VHF-AM en la banda aeronáutica (Figura 2.3)

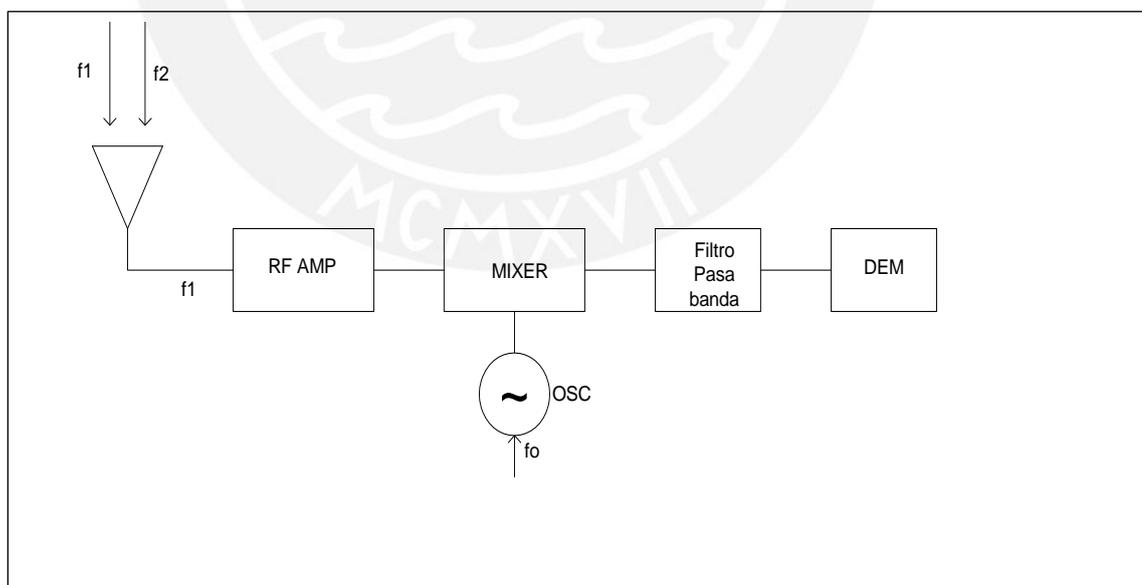


Figura 2.3: Diagrama de bloques de Receptor VHF AM

2.5.1.1 Amplificador de Radiofrecuencia (RF AMP)

Esta es la etapa que se acopla directamente a la antena, en esta etapa se produce distorsión de la señal debido a la *alinealidad* del amplificador. Debido a esta alinealidad propia del amplificador se pueden producir los siguientes fenómenos con las señales:

Intermodulación:

La Intermodulación se produce debido a la combinación de frecuencias presentes en la entrada de RF del amplificador provenientes de la antena en la forma que sigue: $f_{salida} = mf_1 \pm nf_2$ donde m y n son enteros

Los amplificadores de Radiofrecuencia tienen una curva de linealidad la cual dependiendo del orden genera señales no deseadas.

- Si la curva es de segundo orden generara armónicos.
- Si la curva es de tercer orden generara términos indeseables de Intermodulación.

Observación: Cuando la señal es intensa o de gran amplitud, el Amplificador de Radiofrecuencia trabajará en la zona lineal. En este caso los armónicos y términos de Intermodulación son despreciables.

Modulación Cruzada:

Este tipo de modulación es llamado también en los sistemas de comunicaciones como *crosstalk*. Ocurre cuando el audio de una frecuencia portadora f_1 se demodula en un receptor que tiene frecuencia de trabajo f_2 . Este fenómeno se puede analizar con un estudio de Intermodulación.

2.5.1.2 Mezclador (MIXER)

Esta etapa del receptor de radiofrecuencia es también es alineal. En esta etapa además de la Intermodulación anterior se puede analizar los valores de los armónicos: **2f, 3f, etc.**

Los equipos de radiofrecuencia que tienen las características de *Modulación de tercer orden* (3IP) tienen mejor respuesta en la recepción.

Analizando un ejemplo tenemos:

Tenemos un mezclador con las siguientes señales. Ver Figura 2.4

En el dominio de la frecuencia se obtiene:

$$S1(\omega).S2(\omega)=f1(\omega) + f2(\omega)+...$$

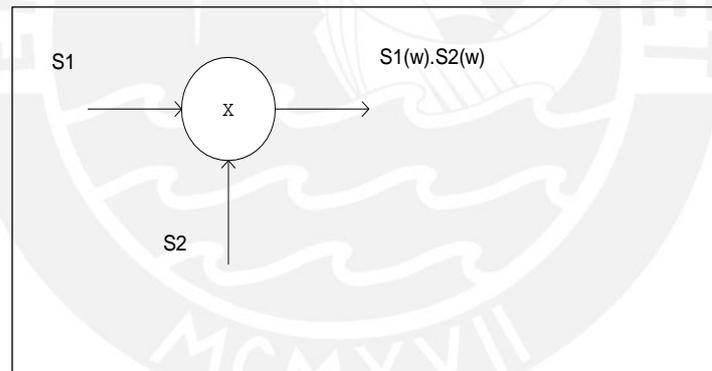


Figura 2.4: Sumatoria de dos señales

Entonces, detallando en la figura 2.5 tenemos para los casos más críticos en el que se tiene una señal $f1$ y una señal interferente $f2$:

1. $f2+f_{mod2}-f2+f1=f1+f_{mod2}$
2. $f2-f_{mod2}-f2+f1=f1-f_{mod2}$
3. $f2-f2+f1=f1$

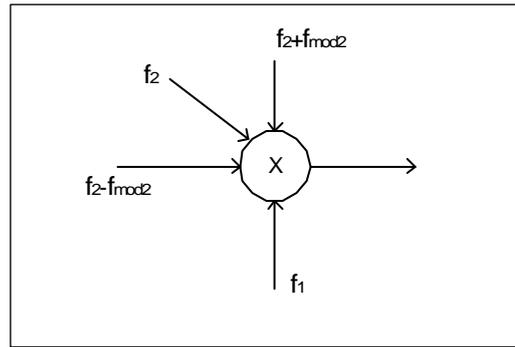


Figura 2.5: Sumatoria de dos señales más interferencia

Por alinealidad tenemos entonces que si en la Recepción tenemos como frecuencia de recepción f_1 y a f_2 como frecuencia interferente, se demuestra matemáticamente que el fenómeno de modulación cruzada puede ocurrir en un caso específico como el analizado es decir se obtiene el audio de la portadora f_2 en la portadora f_1 .

Para solucionar este tipo de problemas que se generan en estos sistemas de comunicaciones se tiene las siguientes alternativas:

- ✓ Instalar un Filtro de cavidades a la entrada del amplificador de radiofrecuencia.
- ✓ Apantallamiento de antenas, esto significa separarlos físicamente.

3. SISTEMAS DE POSICIONAMIENTO GEOGRAFICO

3.1 Sistemas Geodésicos Satelitales

Los sistemas geodésicos satelitales se encuentran definidos mediante orbitas de los satélites sobre sistemas de referencia terrestres.

Hasta la fecha existen cuatro elipsoides geocéntricos definidos:

WGS-60

WGS-66

WGS-72 (Llamado Doppler Transit)

WGS-84 (Llamado elipsoide GPS)

Sistema de Referencia WGS-84

El Sistema Geodésico Mundial WGS 84, es uno de los mejores sistemas terrestres convencionales que representa la forma y tamaño de la tierra.

Las coordenadas de los satélites y usuarios que se posicionan con el sistema GPS *Global Positioning System*, tiene como referencia el sistema WGS-84 (Sistema Geodésico mundial de 1984). Estas coordenadas pueden ser cartesianas en el espacio respecto al centro de masas de la tierra (X,Y,Z) o geodésicos (φ , λ , h).

El sistema tiene las siguientes características, tal como se puede ver en la figura 3.1:

- Origen en el centro de masas de la tierra.
- El eje Z es paralelo al polo medio

- El eje X es la intersección del meridiano de Greenwich y el plano del Ecuador.

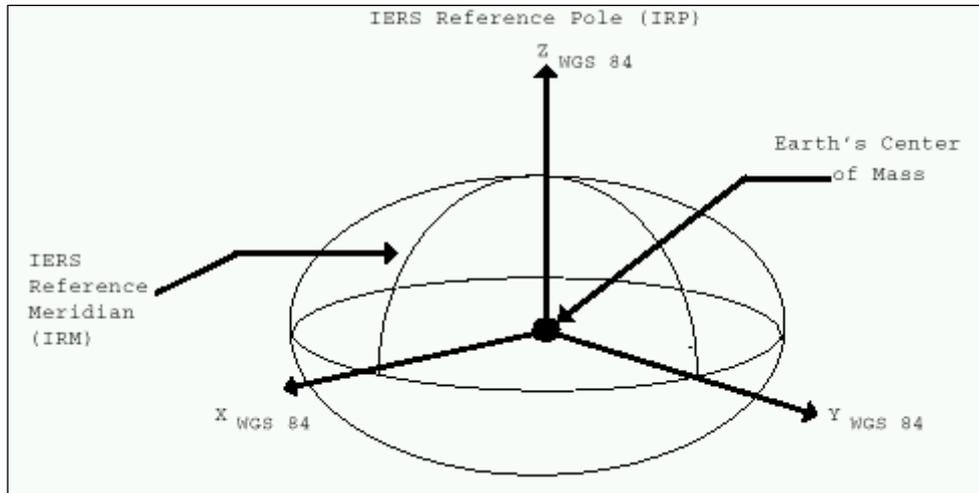


Figura 3.1: Sistema de coordenadas WGS84

El eje Y es perpendicular a los ejes Z y X y coincide con ellos en el centro de masas de la tierra.

IERS: Servicio Internacional de Rotación de la Tierra es el ente responsable del mantenimiento de los estándares globales de tiempo y de los ejes de referencia de la tierra.

EL Concejo de la OACI *Organización de la Aviación Civil Internacional* en sesión del 3 de marzo de 1989 aprobó la Recomendación 3.2/1 en el cual adoptan el Sistema Geodésico Mundial - 1984 (WGS 84) como el estándar geodésico de referencia para los futuros sistemas de navegación respectivos a la aviación civil internacional.

3.2 Sistema de Posicionamiento Global (GPS)

El sistema GPS fue creado por el departamento de defensa de los Estados Unidos (DoD) con la finalidad de constituir un sistema de navegación preciso con fines militares.

Para ello aprovecharon las condiciones de propagación de las ondas de radio de la banda L en el espacio, así como la posibilidad de modular las ondas con la información necesaria que permita posicionar un objeto en cualquier parte del mundo las 24 horas del día.

La banda L del espectro es la que presenta mejor transparencia atmosférica, lo cual es muy importante para la precisión del sistema.

Las dos frecuencias portadoras son denominadas L1 (1.575'42 MHz) y L2 (1.227'60 MHz). El empleo de dos frecuencias distintas se debe a que la atmósfera proporciona un cierto retardo en la propagación de las ondas, siendo este retardo una función de la frecuencia. Al utilizar dos frecuencias distintas se puede conocer ese retardo y compensarlo en consecuencia.

El Sistema GPS está compuesto por tres segmentos: espacial, usuarios y control.

Segmento Espacial

Este segmento está conformado por 24 satélites ubicados a muy alta órbita y que es llamada la constelación *NAVSTAR* (Navegación por satélite en tiempo y distancia). La constelación está formada por seis planos orbitales y en cada uno de ellos existe una órbita elíptica donde se alojan los satélites distribuidos regularmente. Los planos tienen una inclinación de 55° respecto

al plano del Ecuador lo que le permite tener una buena cobertura de los polos. Cada orbita contiene al menos 4 satélites. Los satélites se sitúan a 20200 Km. respecto del Geocentro y completan una orbita en 12 horas.

De esta forma se garantiza la presencia de al menos cuatro satélites sobre el horizonte en cualquier lugar de la superficie terrestre.

El tiempo utilizado por el GPS es el UTC (Tiempo Universal Coordinado) que es definido por el Observatorio Naval de los Estados Unidos usando precisos relojes atómicos de hidrógeno. La unidad del tiempo GPS es el segundo atómico internacional y tiene su origen coincidente con el UTC a las cero horas del 6 de Enero de 1980.

Entre las características de los satélites tenemos:

- Antenas emisoras de ondas de radio en la banda L: para transmitir información al usuario
- Antenas emisoras – receptoras de ondas de radio en la banda S: que permiten que el satélite pueda ser controlado.
- Paneles solares que le proveen energía necesaria para su funcionamiento.
- Reflectores láser para el seguimiento desde el sector de control.

Segmento de Usuarios

Este sector lo componen los instrumentos que un usuario necesita para la recepción, lectura, tratamiento y configuración de las señales con el fin de obtener en este caso una posición geográfica. Los elementos son el equipo de observación y el software asociado.

Equipo de observación: Lo componen la antena, el sensor y la unidad de control.

La antena de recepción recibe las radiaciones electromagnéticas que emiten los satélites y las transforma en impulsos eléctricos los cuales poseen la información modulada en las portadoras.

El sensor recibe los impulsos de la antena receptora y reconstruye e interpreta la señal, es decir, las portadoras, los códigos y el mensaje de navegación. Demodula la señal.

El controlador almacena datos, gestiona la observación y controla el sensor.

Tipos de Receptores

De Navegación: Usados en navegación, catastro y levantamiento de escalas.

Monofrecuencia: Usados en aplicaciones topográficas y geodésicos en distancias pequeñas de hasta 100 Km.

Bifrecuencia: Usados para trabajos de precisión.

Segmento de Control

Son todos los elementos que se utilizan para controlar los satélites, instalados en las estaciones de Control. El segmento de control consiste en un sistema de estaciones de seguimiento, localizadas alrededor del mundo.

La estación maestra de control (MCS) está situada en Falcón AFB en Colorado Spring. Las estaciones de control miden las señales procedentes de los satélites y son incorporadas en modelos orbitales para cada satélite. Los modelos calculan datos de ajuste de órbita (efemérides) y correcciones de los relojes de cada satélite. La estación maestra envía, a cada satélite, las efemérides y

correcciones de reloj. Cada satélite envía, posteriormente, y mediante señales de radio, subconjuntos de estas informaciones a los receptores de GPS.

3.3 Diferencial GPS

El sistema GPS diferencial consiste en fijar un punto de referencia o estación central desde donde se envía por ondas de radio en la banda de VHF a los usuarios el valor que deben corregir en su posición. Este valor se calcula con la diferencia que existe entre una medida de un receptor GPS de un punto determinado con el valor real que se tiene de mediciones topográficas. Este sistema permite obtener medidas más exactas de la posición. Se utiliza para aplicaciones de precisión como por ejemplo aterrizaje de aeronaves por instrumentos.

3.4 PROTOCOLO NMEA 0183

NMEA es un protocolo de transmisión de datos que se usa para la navegación marítima tanto como terrestre. Ha sido definido, y está controlado, por la organización estadounidense *National Marine Electronics Association*

Una vez que un equipo receptor GPS sabe donde está (es decir conoce las coordenadas geográficas de su posición), es posible suministrar dicha información mediante alguna interfase a otro dispositivo como puede ser un computador (o cualquier otro tipo de equipo especializado como un microcontrolador). La información de tipo NMEA se ha estandarizado a nivel mundial. Además de nuestra posición suministra información de la dirección de nuestro desplazamiento, nuestra velocidad actual, cual es el *waypoint* (lugar

prefijado) de destino, qué satélites esta recibiendo, la intensidad de las señales que se reciben, la posición de los satélites (si están hacia el norte, sur, etc.), el *datum* que se está usando. Estas sentencias son un conjunto de "frases" que contienen la información que acabamos de describir anteriormente. Hay muchos programas que te permiten ver como son esas sentencias. Normalmente esta información la envía el equipo GPS a través de un puerto serial.

Las principales sentencias que comprende este protocolo son las siguientes:

- \$GPAAM - Waypoint Arrival Alarm
- \$GPASD - Autopilot System Data
- \$GPBOD - Bearing, Origin to Destination
- \$GPBWW - Bearing, Waypoint to Waypoint
- \$GPGGA - Global Positioning System Fix Data
- \$GPGLL - Geographic Position, Latitude/Longitude
- \$GPGSA - GPS DOP and Active Satellites
- \$GPGSV - GPS Satellites in View
- \$GPMTA - Air Temperature (to be phased out)
- \$GPMWD - Wind Direction
- \$GPMWV - Wind Speed and Angle
- \$GPRTE - Routes
- \$GPVBW - Dual Ground/Water Speed
- \$GPVPW - Speed, Measured Parallel to Wind
- \$GPWCV - Waypoint Closure Velocity
- \$GPWNC - Distance, Waypoint to Waypoint
- \$GPWPL - Waypoint Location

\$GPXTE - Cross-Track Error, Measured

\$GPZFO - UTC & Time from Origin Waypoint

\$GPZTG - UTC & Time to Destination Waypoint

Tenemos un ejemplo con la sentencia GPGGA del significado de cada campo de esta sentencia.

La sentencia NMEA sería una como esta:

*\$GPGGA,161555,4321.1752,N,00823.0447,W,2,08,02.3,0009,M,-053,M,02,0508*4F*

El significado de cada campo es:

\$GPGGA..... Identifica el tipo de la sentencia.

161555..... horas, minutos y segundos

4321.1752, N.... latitud Norte

00823.0447, W... longitud Oeste

2..... Calidad de la posición: 2 posiciones por DGPS

08..... Número de satélites monitorizados

2.3.....Degradación horizontal de la posición

0009,M.....Altitud en metros sobre el nivel medio del mar

-053,M.....Altura del Geoide sobre el elipsoide WGS84

02.....Intervalo en segundos desde la última posición

0508.....Número de identificación de la estación

EL valor *4F* es un valor de comprobación, depende de los valores anteriores



4. PROTOCOLOS DE TRANSMISION DE DATOS

Los protocolos de transmisión de datos son un conjunto de reglas que posibilitan el intercambio de información entre dos o más estaciones transmisoras y/o receptoras. La complejidad de un protocolo radica en dos aspectos: el número de estaciones involucradas en la comunicación a través de un medio de transmisión y el método de acceso al canal.

4.1 Características de los protocolos:

Control de errores: Debido a que en todos los sistemas de comunicación cabe la posibilidad de que aparezcan errores por la distorsión de la señal transmitida en el camino que va desde el emisor al receptor, se hace necesario el uso de control de errores; a través de un procedimiento de detección y corrección de errores (descarte o retransmisión de los datos).

Para el control de errores se utilizan unas técnicas necesarias para recuperar pérdidas o deterioros de los datos y de la información de control. Por lo general el control de errores se aplica por medio de dos funciones separadas: La retransmisión y la detección de errores.

Control de flujo de datos: Para evitar que el emisor sature al receptor transmitiendo datos más rápido de lo que el receptor o destino pueda asimilar y procesar, se hace necesario el uso de ciertos procedimientos llamados controles de flujo.

El control de flujo es una operación realizada por el receptor (destino) para limitar la velocidad o cantidad de datos que envía la entidad el emisor (origen o fuente). Una de las maneras de aplicar el control de flujo es

mediante el uso de “parada y espera”, en el que se debe confirmar el paquete de información recibido antes de enviar el siguiente.

Otra manera de utilizar un control de flujo es mediante el envío de la información de la cantidad de datos que pueden ser transmitidos sin tener que esperar la confirmación.

Formato de los datos: Esto tiene que ver con el acuerdo que debe existir entre las dos partes respecto al formato de los datos intercambiados, como por ejemplo el código binario usado para representar los caracteres.

Orden de los datos: El orden de los datos es esencial en una red donde existen diferentes estaciones conectadas, debido a que los paquetes de información pueden ser recibidos de manera diferente, ya sea por que toman caminos distintos a través de la red, por ejemplo, si el paquete 1 toma una ruta larga y el paquete 2 toma una corta, evidentemente el paquete No. 2 llegará primero (suponiendo que los dos paquetes son del mismo tamaño), y los datos recibidos no serán los mismos que están en el emisor (debido a que tendrán un orden diferente en el receptor).

4.2 Protocolos Orientados a Bit: son aquellos protocolos en los cuales los bits por si solos pueden proveer información, son protocolos muy eficientes y trabajan en tramas de longitud variable.

Un protocolo básico comprende varios campos de información que son los siguientes:

- Bandera o *flag*: Se utilizan al principio y al final de la trama para sincronizar el sistema.
- Campo de dirección: Es una secuencia de bits que identifica las estaciones en una comunicación.
- Campo de control: Es una secuencia de bits que permite establecer comandos o respuestas.
- Campo de datos: Contiene toda la información.
- Campo de Chequeo de Errores: Es un algoritmo que permite el chequeo de errores.

4.3 Protocolos Orientados a Byte: son aquellos protocolos en los cuales la información viene provista por la conjunción de bytes de información y bytes de control.

Los protocolos orientados a byte son aquellos en que las reglas y funciones son establecidas a partir de caracteres especialmente designados para tal. Los más difundidos son el *Start/Stop* y el *BSC* que aun son utilizados en redes de datos de todo el mundo. El *BSC*, además, es un protocolo asíncrono.

Los protocolos orientados a byte o caracter utilizan bytes completos para representar códigos de control establecidos tales como los definidos por el código ASCII (*American Standard Code for Information Interchange*) o código EBCDIC (*Extended Binary Coded Decimal Interchange Code*)

4.4 Protocolos Asíncronos

Son aquellas transmisiones en el que cada trama de datos transmitido incluye bits de arranque, bits de parada al principio y al final. La función principal de estos datos es la de avisar al receptor de que esta llegando un dato y además de darle un tiempo suficiente al receptor de realizar funciones de sincronismo antes de que llegue la siguiente información.

4.5 Protocolos Síncronos

En este tipo de transmisiones se utilizan canales separados de reloj que administran la recepción y transmisión de los datos. Al inicio de cada transmisión se emplean unas señales preliminares denominadas:

Bytes de sincronización en los protocolos orientados a byte.

Banderas en los protocolos orientados a bit.

Su misión principal es alertar al receptor de la llegada de los datos.

En este caso las señales del reloj determinan la velocidad a la que se realizan las transmisiones

5. DISPOSITIVOS ELECTRONICOS

5.1 Amplificadores Operacionales

Los amplificadores operacionales también llamados *opamps*, son circuitos integrados compuestos de hasta cientos o miles de transistores que permiten la amplificación y manipulación de señales eléctricas. Sus usos pueden ser variados, pasando desde sumadores, restadores, multiplicadores, integradores, derivadores, hasta funciones exponenciales, divisiones y muchas más.

Pueden ser manipulados en lazo abierto, presentando ganancias típicas de 5000 o más, o en lazo cerrado en donde la ganancia estará definida por los elementos externos que se le conecten.

La figura 5.1 permite ver los dos tipos de configuraciones y sus ganancias.

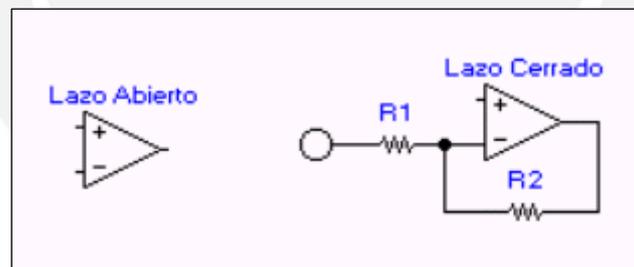


Figura 5.1: Configuración básica de OPAMPS

A.) Ganancia en Lazo abierto: $A = 5000$ o mas

B.) Ganancia en Lazo cerrado: $A = R2/R1$

Los amplificadores operacionales se pueden configurar en diferentes modos de operación, los principales son: Inversor, No Inversor, Diferencial.

Amplificador Inversor:

El circuito amplificador de ganancia constante que más se utiliza es el amplificador inversor, el cual se muestra en la figura 5.2. La salida se obtiene multiplicando la entrada V_1 por una ganancia fija o constante, que determinan la resistencia de entrada (R_1) y la resistencia de retroalimentación (R_f), con la salida invertida respecto a la entrada.

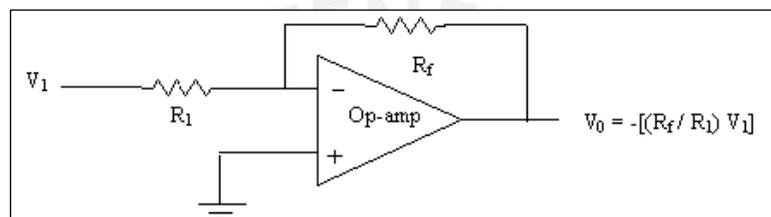


Figura 5.2: Amplificador Inversor

Como observación podemos notar que si R_f es igual a R_1 tendremos un circuito amplificador con ganancia unitaria. Ecuación: $V_0 = -[(R_f / R_1) V_1]$

Amplificador No Inversor

Las conexiones de la figura muestran un circuito *opamp* que trabaja como amplificador no inversor o multiplicador de ganancia constante. Debe resaltarse que la configuración amplificador inversor es la que más se utiliza, porque tiene una mejor estabilidad de frecuencia.

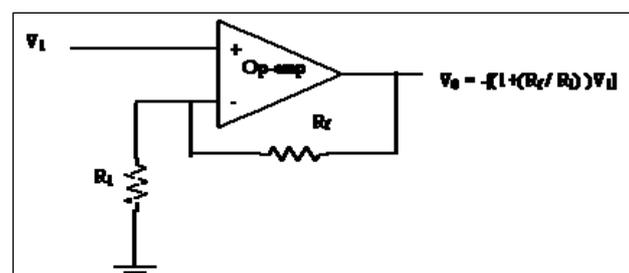


Figura 5.3: Amplificador No Inversor

Se puede deducir la siguiente formula asumiendo que V_1 es el voltaje en R_1 es decir V_- .

$$V_1 = \frac{R_1}{R_1 + R_f} V_o$$

Lo que da como resultado

$$\frac{V_o}{V_1} = \frac{R_1 + R_f}{R_1} = 1 + \frac{R_f}{R_1}$$

Amplificador Seguidor de Voltaje

El circuito seguidor unitario, que se muestra en la figura proporciona una ganancia unitaria sin inversión de polaridad o fase, está claro que: $V_o = V_1$, y que la salida es de la misma polaridad y magnitud que la entrada. El circuito opera como un circuito emisor seguidor o seguidor de fuente, a excepción de que la ganancia es exactamente unitaria

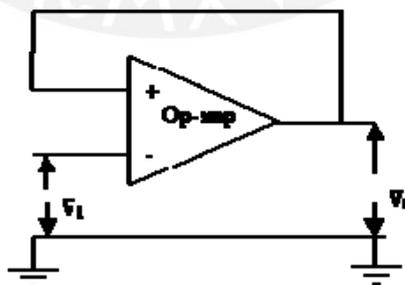


Figura 5.4: Amplificador Seguidor de Voltaje

Este circuito se puede usar como un circuito adaptador de impedancias entre dos circuitos diferentes como pueden ser una entrada de audio de un radio o amplificador y una salida de un generador de señales de audio.

5.2 Transistores

Un transistor es un dispositivo semiconductor de estado sólido con tres terminales y que es utilizado como amplificador o interruptor en el que una pequeña corriente o tensión aplicada a uno de los terminales controla o modula la corriente entre los otros dos terminales. Es el componente fundamental de la moderna electrónica, tanto digital como analógica. En los circuitos digitales se usan como interruptores, y disposiciones especiales de transistores configuran las puertas lógicas, memorias RAM y otros dispositivos; en los circuitos analógicos se usan principalmente como amplificadores.

El transistor bipolar (conocido universalmente con la simple denominación de transistor) es un circuito de tres terminales que puede cumplir funciones de amplificador (operación lineal en región activa directa) o conmutador (operación en la zona de corte y saturación). En la figura 5.5 tenemos los símbolos del transistor bipolar: tipo NPN y tipo PNP

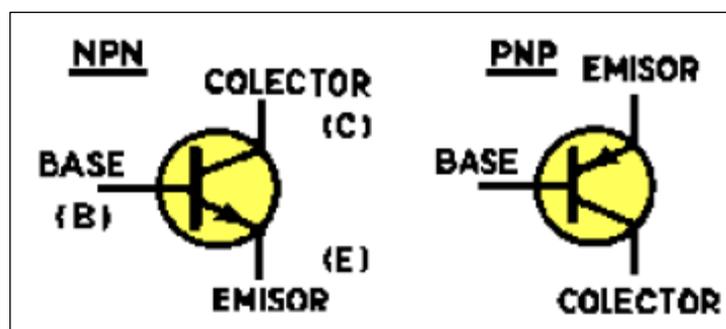


Figura 5.5: Símbolos de transistores bipolares

Vamos a analizar el funcionamiento del transistor como un conmutador que es el caso que se usa en el presente proyecto. Los dos casos posibles de operación son los siguientes:

Caso de Corte: Si la corriente de la base B es nula entonces la corriente en el colector C será prácticamente cero. Esto significa que el circuito entre el colector y el emisor está abierto y no circulará corriente por estas junturas.

Caso de Saturación: Si la corriente en la base es significativa entonces circulará corriente por el colector hacia el emisor, si saturamos la base con la suficiente corriente entonces la tensión entre el colector y el emisor se acercará a cero y esta unión se comportará como si fuera un cable es decir en cortocircuito. A este estado se le conoce como saturación del transistor.

En ambos casos (transistor en corte o saturado), la potencia disipada entre colector y emisor es mínima: en el primer caso es obvio, ya que no circula corriente; en el segundo si bien implica una circulación de corriente máxima, coincide con una tensión colector-emisor cercana a cero (el producto de la tensión por la corriente es reducido).

6. MULTICANALIZACION Y ACCESO AL MEDIO

6.1 Multicanalización

6.1.1 TDMA (Acceso Múltiple por División de Tiempo)

Es un método de multicanalización en el dominio del tiempo; esto significa que si queremos enviar 3 canales por un mismo medio físico y a la misma frecuencia haciendo uso de TDMA, lo que se hace es asignar una duración determinada de tiempo a cada canal, y se les cederá el medio físico a cada canal durante ese espacio de tiempo determinados denominados ranuras de tiempo o *timeslots*. Este sistema requiere métodos de sincronismo muy eficaces. Ver figura 6.1

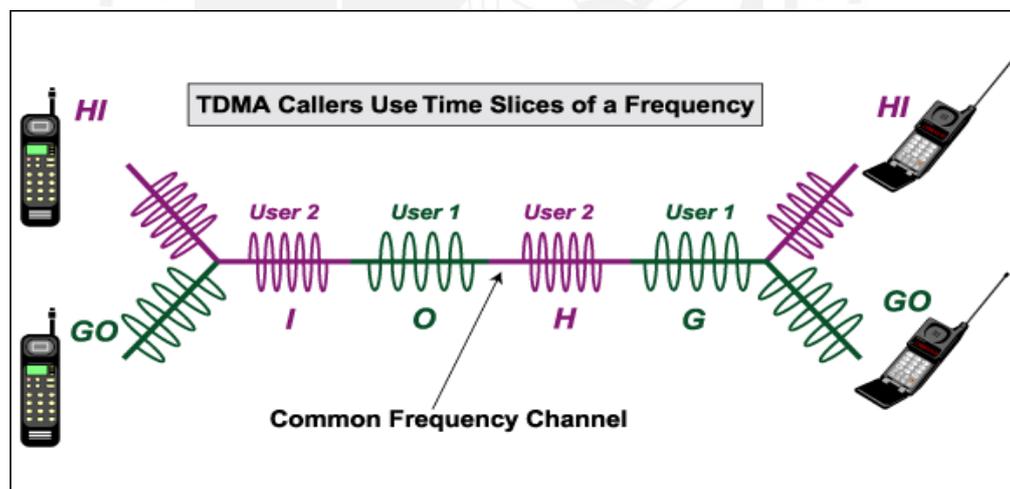


Figura 6.1: Diagrama TDMA

6.1.2 FDMA (Acceso Múltiple por División de Frecuencia)

En este caso haciendo uso de modulaciones enviamos cada canal en una banda de frecuencias distinta. Luego en cada receptor se debe demodular para devolver la transmisión a banda base, o a su banda natural. Es un sistema

ampliamente usado en radiocomunicaciones. A cada usuario se le asigna una banda de frecuencia exclusiva. Ver figura 6.2

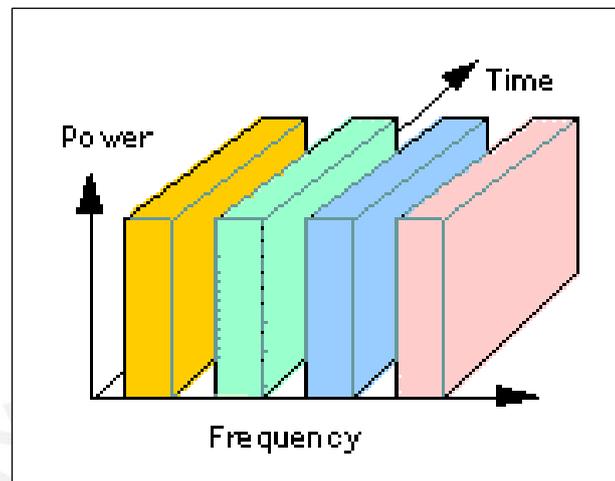


Figura 6.2: Diagrama FDMA

6.2 Métodos de Acceso al Medio

El acceso al medio se refiere al procedimiento en el que un elemento de una determinada red de comunicaciones toma el canal de transmisión para usarlo.

(Ver figura 6.3)

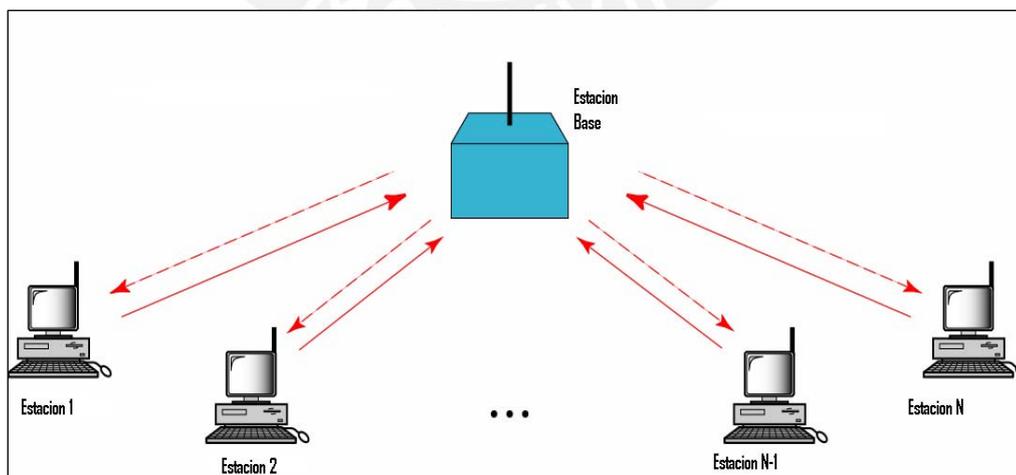


Figura 6.3: Diagrama de acceso al medio

El protocolo CSMA *Acceso Múltiple por Detección de Portadora* es uno de los protocolos más populares en los distintos tipos de redes. En el caso de redes de área local por ejemplo, en los canales de *broadcast* con pequeños retardos en la propagación de las señales, las colisiones en el canal pueden ser reducidas considerablemente haciendo que cada estación sondee el canal para detectar la presencia de alguna portadora en el canal de transmisión antes de acceder a dicho canal. Seguido de una transmisión exitosa cada estación que se encuentra lista transmite con la misma probabilidad de 100% en el siguiente periodo de tiempo. De darse el caso, luego de detectar una colisión cada estación ejecuta un algoritmo *adaptivo* para seleccionar la probabilidad de transmitir en el siguiente periodo de tiempo. El proceso de sondeo del canal no es por demanda, cada estación está equipada con un receptor, sin embargo la detección de la portadora no nos libera de una colisión. Algunas variaciones de este protocolo incluyen 1-persistente, 0-persistente, y p-persistente CSMA.

6.2.1 CSMA/CD

El CSMA/CD es un método de transmisión de datos en donde todas las estaciones pueden mandar datos con una señal eléctrica sumada (portadora). En caso de que existan transmisiones simultáneas detectan las colisiones. Es la base de la topología Ethernet.

El protocolo CSMA-CD funciona de la siguiente manera: un nodo que desea transmitir espera a que el canal esté aislado, una vez que se encuentra en este estado empieza la transmisión. Si otro nodo empezara también a transmitir en este instante se produciría colisión, por lo tanto se detiene la transmisión y se

retransmite tras un retraso aleatorio. Para el análisis se define CBD Channel Busy Detect como un indicador o bandera que sondea la presencia de portadora.

Tipos de CSMA

- i. **Aloha Puro:** En este caso, el protocolo ignora cualquier indicador CBD y transmite cuando tenga alguna información que transmitir.

El algoritmo es como sigue: Primero fija el valor de un contador denominado *backoff* en cero luego transmite la trama por el canal y espera una respuesta de reconocimiento, si esta es recibida entonces la transmisión de esa trama fue exitosa. Si no se recibe el reconocimiento se incrementa el *backoff*, se compara este con un valor límite, si es menor esperamos un tiempo igual al *backoff* y vuelve a transmitir, si el *backoff* llegó al valor límite se aborta la transmisión.

- ii. **Slotted Aloha:** Para caso, el protocolo ignora el indicador CBD y transmite durante un periodo limitado cuando tenga alguna información que transmitir. El algoritmo de transmisión es similar al del Aloha Puro, con la diferencia que antes de transmitir espera que empiece su ranura de tiempo asignada.
- iii. **1-Persistente CSMA:** En este caso, el protocolo CSMA tan pronto como detecte el canal libre, transmite la información que tenga que transmitir.
- iv. **P-Persistente CSMA:** En este caso, el protocolo tan pronto como tenga el canal libre transmite el P% del tiempo, espera un tiempo definido y vuelve a correr el algoritmo predefinido.

- v. **No-Persistente CSMA:** En este caso, tan pronto como el algoritmo detecte el canal libre, espera un tiempo aleatorio (entre 30 y 300milisegundos) y vuelve a correr el algoritmo.

El uso de 1-Persistente y P-Persistente ($P > 10$) puede incrementar las colisiones y reducir la efectividad del *throughput* de la transmisión. Los protocolos CSMA Persistente y no persistente son una mejora del protocolo Aloha.

Es decir, que en el protocolo CSMA si el canal es detectado como ocupado ninguna estación transmite.

En el protocolo CSMA con detección de colisiones (CSMA/CD), las estaciones abortan la transmisión cuando ellos detectan una colisión. Este caso puede ocurrir cuando dos estaciones al mismo tiempo detectan que el canal esta libre y empiezan a transmitir simultáneamente.

6.2.2 CSMA/CA: Protocolo de acceso al medio

El algoritmo básico de acceso a este nivel es muy similar al implementado en el estándar IEEE 802.3 y es el llamado CSMA/CA (*Carrier Sense Multiple Access / Collision Avoidance*). Este algoritmo funciona tal y como se describe a continuación:

1. Antes de transmitir información una estación debe testear el medio, o canal inalámbrico, para determinar su estado (libre / ocupado).
2. Si el medio no esta ocupado por ninguna otra trama la estación ejecuta una espera adicional llamada espaciado entre tramas (IFS).

3. Si durante este intervalo temporal, o bien ya desde el principio, el medio se determina ocupado, entonces la estación debe esperar hasta el final de la transacción actual antes de realizar cualquier acción.

4. Una vez finalizada esta espera debida a la ocupación del medio la estación ejecuta el llamado algoritmo de Backoff, según el cual se determina una espera adicional y aleatoria escogida uniformemente en un intervalo llamado ventana de contienda (CW). El algoritmo de Backoff nos da un número aleatorio y entero de ranuras temporales (slot time) y su función es la de reducir la probabilidad de colisión que es máxima cuando varias estaciones están esperando a que el medio quede libre para transmitir.

5. Mientras se ejecuta la espera marcada por el algoritmo de Backoff, se continúa escuchando el medio, de tal manera, que si el medio se determina libre durante un tiempo de al menos 1 IFS, esta espera va avanzando temporalmente hasta que la estación consume todas las ranuras temporales asignadas.

En cambio, si el medio no permanece libre durante un tiempo igual o superior a IFS, el algoritmo de Backoff queda suspendido hasta que se cumpla esta condición. Cada retransmisión provocará que el valor de CW, que se encontrará entre Cw_{min} y Cw_{max} se duplique hasta llegar al valor máximo. Por otra parte, el valor del slot time es 2000 Seg. En la siguiente página en la figura 6.4 se puede ver un ejemplo de CSMA/CA.

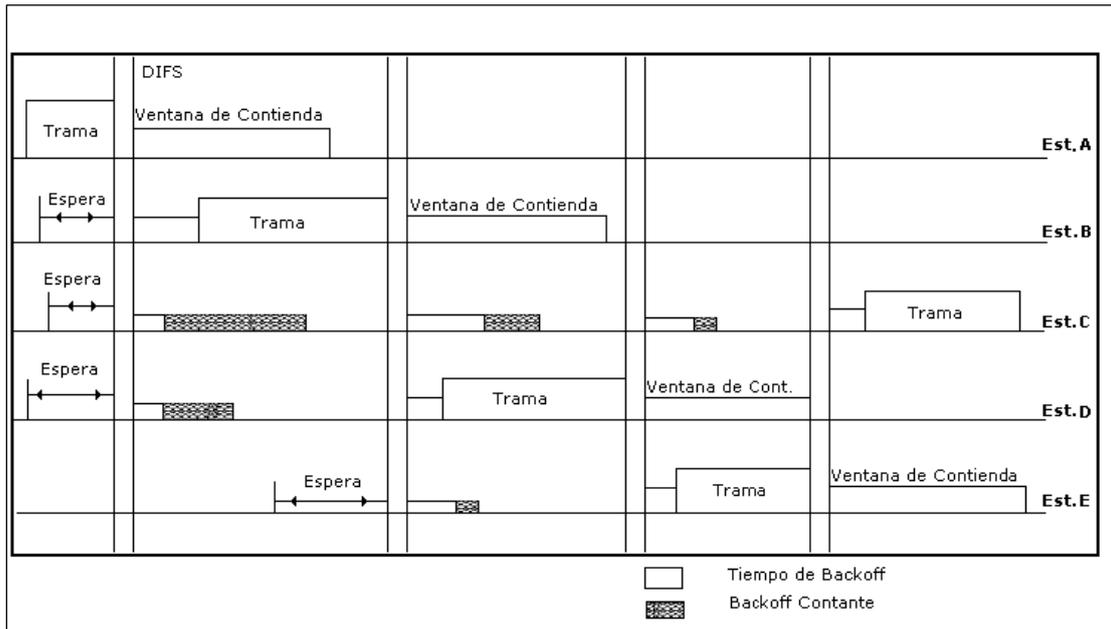


Figura 6.4: Diagrama de CSMA/CA

7. DESARROLLO DE LA APLICACIÓN

En este capítulo vamos a detallar todos los procedimientos necesarios para implementar un sistema de transmisión de señales de frecuencias audibles a través de radios de VHF/AM, como una aplicación de intercambio de información de posición geográfica entre una aeronave y una estación base. Se detalla desde el análisis de los dispositivos que se usan, el desarrollo de los programas en Lenguaje Turbo C/C++ y en lenguaje ensamblador, el diseño de los circuitos, análisis de la forma de acceso al medio y las pruebas que se realizaron en el Laboratorio y en el campo.

7.1 Estudio de la forma de modulación digital

Existen varias formas de modular información digital para transmitirla por determinado ancho de banda de un canal. En este caso en especial, la principal característica que encontramos en las radios *VHF-AM* convencionales usadas es que la señal moduladora debe trabajar en el rango de audio y según las especificaciones técnicas tiene un Ancho de Banda de 2.5 KHz. a -3dB (Ver gráfico 7.1), lo que vendría entonces a ser la principal limitación para la transmisión de información.

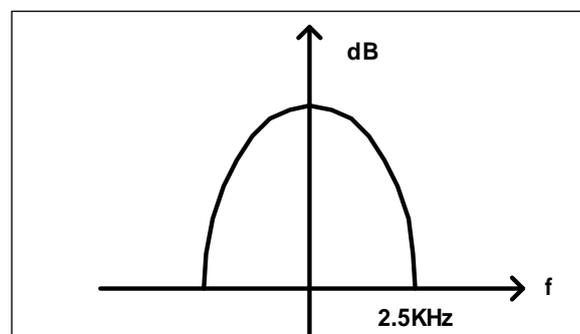


Figura 7.1: Ancho de banda del audio

Analizando las posibilidades existentes, llegamos a la conclusión de que para las pruebas se podrían utilizar los siguientes tipos de modulación: *Frequency Shift Keying (FSK)* o *Dual Tone Multi Frecuencia (DTMF)*.

El análisis del **FSK** nos llevó a la conclusión de que si bien era posible obtener una señal *FSK* cuyo ancho de banda necesario para ser transmitido sea menor o igual al ancho de banda especificado por los equipos de radio, generalmente los IC's *FSK* trabajan a mucha mayor frecuencia y no son comunes en el mercado.

El análisis de la transmisión de la información utilizando tonos duales **DTMF**, nos condujo a que los tonos o frecuencias de audio generados por dicho circuito tenían ya valores fijos y que la máxima frecuencia generada es 1633 KHz. ($BW_{max} DTMF$), de esto se puede inferir teóricamente que no habría ningún problema para la transmisión de estas señales por el ancho de banda de las radios AM (BW_{Canal}). (Ver Grafico 7.2).

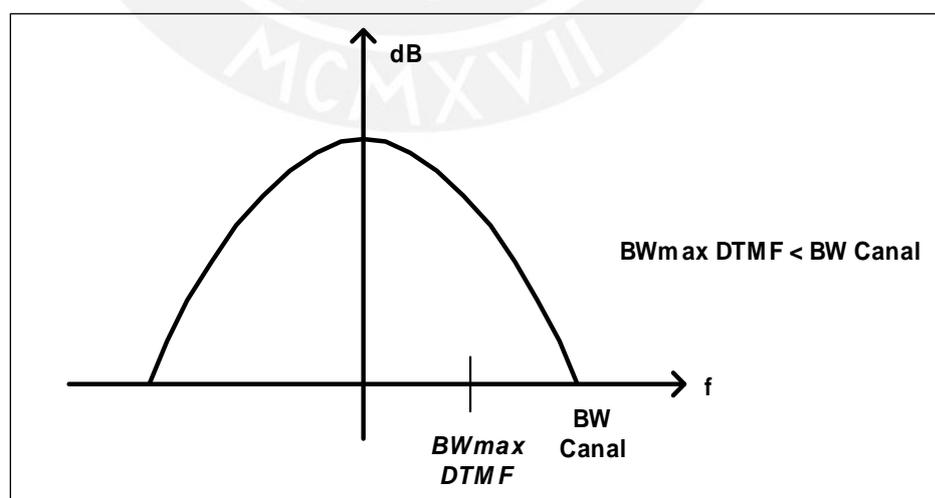


Figura 7.2: Ancho de banda del canal

7.2 Análisis de características de las Radios VHF –AM

Los radios transceptores provistos por Corpac tienen las siguientes características:

Marca: *DX RadioSystems*

Potencia de transmisión: 8 Vatios

Sensibilidad: 0.5 micro voltios.

Banda de frecuencia de uso: banda aeronáutica.

Definimos sensibilidad como el mínimo valor de señal recibida que el receptor es capaz de demodular. Para las pruebas de este proyecto se asignó por parte de CORPAC como frecuencia de trabajo: 135.5 MHz.

Estos sistemas de radio usan un sistema Half-Duplex para transmitir y recibir información. Es decir mientras están transmitiendo ocupan el canal y solo pueden recibir cuando dejan de transmitir.

Se sometió a los radios a pruebas con la finalidad de verificar que estos no distorsionen el tono modulador y para que el receptor DTMF sea capaz de decodificar la información enviada y recibida a través de los radios.

Para esto se midió la distorsión que introducen los radios, con un equipo Analizador de Radiocomunicación de marca *Hewlett-Packard*, con los siguientes parámetros:

Tono generado: frecuencia 1KHz Amplitud 100 mV

Obtenemos 1.9% de distorsión y 13.2 dB de SINAD.

Definimos SINAD como el valor de Señal a Ruido más Distorsión, expresado en decibeles.

Según las especificaciones técnicas de los radios, estos son valores que están dentro del rango de operación normal del equipo. De esta forma descartamos un posible error que se pueda introducir en los radios: que al pasar los tonos audibles a través de los radios, estos puedan distorsionar dicha señal.

Se empezó a trabajar con los tonos duales DTMF, para esto se selecciona el generador de tonos DTMF Mitel 5089 y el decodificador de tonos DTMF Mitel 8870.

La primera prueba que se hizo fue generar un tono DTMF y enviarla por el transmisor VHF-AM, luego recibir la señal de un receptor VHF-AM y conectarla al receptor o decodificador de tonos y comprobar si decodifica los valores enviados. Con este fin se diseña un programa en Turbo C/C++ y un protocolo de transmisión de datos que maneje el envío y recepción de tramas.

7.3 Diseño del protocolo de transmisión de datos

7.3.1 Análisis de datos recibidos del receptor GPS

Se seleccionó un equipo Receptor GPS marca *TRIMBLE* modelo *FLIGHTMATE PRO GPS* provisto por Corpac, se analizó este equipo y la forma como trabaja.

Existen varios formatos en la configuración del equipo GPS que permiten obtener la información en forma serial. Este equipo trabaja con el estándar NMEA y se escogió la sentencia GPGGA, esta opción nos permite recibir la información, la cual nos es enviada por el puerto serial, los datos recibidos son de Latitud, Longitud, Altitud, Hora y otros.

Se recibe esta información en un programa desarrollado en lenguaje Turbo C/C++ y se almacena en un buffer creado por software, luego se crea

inicialmente una rutina de verificación de la información que llega del GPS de acuerdo al formato de este estándar.

Luego de que esta información es verificada: cabecera e identificador, se extrae del buffer de almacenamiento definido los datos que se van a utilizar para el procesamiento en el siguiente formato:

Latitud: $gg^{\circ} mm.dd''$ Longitud: $gg^{\circ}mm.dd''$.

7.3.2 Diseño de la trama

Se empezó con el desarrollo de la estructura de la trama de información que se emplearía para las pruebas a la que denominamos **ADS Frame** (nombre dado por Corpac).

El objetivo es enviar la posición de un cierto móvil para visualizar su ubicación y movimiento en una pantalla en la estación base. Con la distancia a la estación base y el ángulo que forma con el eje del Norte o azimut se puede obtener la ubicación del punto emisor de información. (*Ver figura 7.3*)

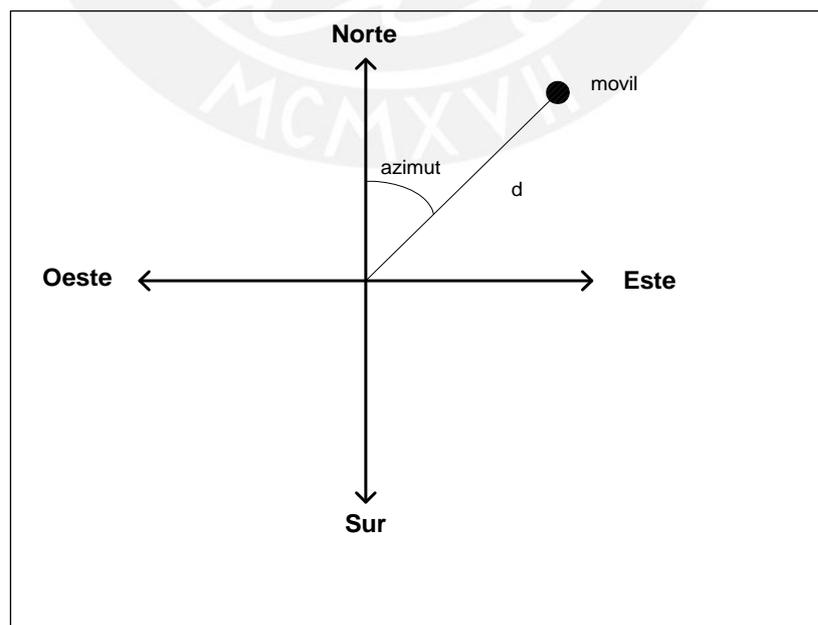


Figura 7.3: Coordenadas cartesianas de ubicación

Analizando la forma como se va a enviar esta información y la cantidad de bits necesarios para transmitir una posición, se descarta el envío de acimut y distancia, porque los rangos de valores son demasiado grandes y sería demasiada información para ensamblar en la trama Ads Frame.

Se analiza optimizar el formato y el algoritmo de ensamble de la trama, se decide que se transmitirá un diferencial de latitud y un diferencial de longitud. Esto significa que la estación base y el móvil deberán estar sincronizados con un punto de referencia (x, y) constante y que deberá ser ingresado por el usuario como un parámetro de entrada del programa. Para optimizar el tamaño de la información, este punto de referencia se hará coincidir con las coordenadas de la estación base, de esta manera disminuimos el tamaño de la información que se va a enviar a simplemente la diferencia de distancia entre el móvil y la estación o punto de referencia.

Esto impone la siguiente restricción: que va a existir un área de cobertura limitada a la cantidad de bits asignada a codificar el valor de la diferencia de ubicación, es decir el programa va a cubrir una determinada zona alrededor del punto de referencia. Si el móvil está fuera de la zona de cobertura no se podrá apreciar su movimiento en la pantalla.

La información que se puede enviar está restringida a bits dado que solo hay 16 tonos DTMF válidos (Ver Tabla 7.1). Con estos valores podemos tomar el código hexadecimal como sistema de codificación de bits para transmitir, esto significa además que cada tono de transmisión corresponderá a 4 bits de información de la trama ADS Frame.

En tal sentido, la trama ADS FRAME se constituye como una estructura orientada a bits.

Frecuencia Baja (Hz)	Frecuencia Alta (Hz)	Digito	D3	D2	D1	D0
697	1209	1	0	0	0	1
697	1336	2	0	0	1	0
697	1477	3	0	0	1	1
770	1209	4	0	1	0	0
770	1336	5	0	1	0	1
770	1477	6	0	1	1	0
852	1209	7	0	1	1	1
852	1336	8	1	0	0	0
852	1477	9	1	0	0	1
941	1336	0	1	0	1	0
941	1209	*	1	0	1	1
941	1477	#	1	1	0	0
697	1633	A	1	1	0	1
770	1633	B	1	1	1	0
852	1633	C	1	1	1	1
941	1633	D	0	0	0	0

Tabla 7.1: Tabla de frecuencias DTMF

Para la trama inicialmente se le dieron los siguientes campos:

Bits

0	1	2	3	4	5	6	7
Id				Δ latitud			
Δ latitud				Δ lat Δ Long			
Δ longitud				Δ longitud			

Especificando tenemos:

- Id Identificador 4 bits
- Δ lat Delta de Latitud 10 bits
- Δ lon Delta de Longitud 10 bits

Para las primeras pruebas de la trama ADS FRAME no se incluyeron campos para verificación de los datos, ni campo de inicio de trama, ya que solamente se desea probar el funcionamiento del algoritmo descrito anteriormente respecto de la transmisión de la diferencia de distancia. Tenemos entonces 24 bits que divididos entre 4 bits nos da 8 tonos que debemos generar.

El algoritmo de Transmisión es como sigue: Tenemos definido el tamaño del diferencial en 10bits lo que equivale a 1024 combinaciones. Luego, definimos el área de cobertura en 15 millas terrestres. Este valor dividido entre 1024 combinaciones nos da un factor igual a 0.0146484375. Luego el valor calculado del diferencial de posición se divide entre el factor lo que nos da un valor denominado Delta de Posición.

Ejemplo: Supongamos el caso en que el diferencial es 14 millas cercano al limite, entonces $14 / 0.0146484375 = 955.73$, este es valor que se codifica en binario para ser transmitido.

En la recepción se realiza el proceso inverso a la transmisión, el valor de Delta se multiplica por el factor y se adiciona la posición de referencia obteniendo la posición actual. Se procedió luego a crear rutinas para el ensamblado y desensamblado de la trama ADS FRAME, dichas rutinas fueron creadas en Lenguaje Turbo C/C++. (Ver anexo numero 3)

7.4 Primera Etapa

En esta primera etapa se procede a diseñar y confeccionar un circuito electrónico de pruebas del sistema, se usa tanto en la transmisión y en la recepción una PC que es la que maneja el programa.

7.4.1 Diseño del Circuito Electrónico

Para la generación de tonos DTMF se seleccionó el circuito Mitel 5089, este circuito es utilizado mayormente en sistemas de señalización entre las centrales telefónicas y los teléfonos. Por ejemplo cuando uno presiona un número cualquiera, se está generando un par de tonos que viajan a través de la línea telefónica y en el otro extremo este par de tonos es decodificado en el número presionado. En el caso del 5089, este posee 8 entradas paralelas digitales (TTL compatibles), lo que nos da una posible combinación de 256 valores. Sin embargo el sistema DTMF permite únicamente 16 posibles combinaciones de pares de tonos validos: 1,2,3,4,5,6,7,8,9,0,#,*,A,B,C,D .

Usando este mismo principio, se hizo un programa donde se generan *nibbles* (4 bits) de información que serán transmitidos en forma de tonos DTMF.

Por principio de funcionamiento del circuito, los tonos deben tener cierto tiempo de duración, el estándar es 50 milisegundos de presencia de tono y 50 milisegundos de ausencia de tono en el canal de transmisión para transmitir un *nibble*, esto como una característica técnica del circuito y así se de el correcto funcionamiento de los circuitos integrados (*Ver figura 7.4*). Esto se logra polarizando correctamente tal como la hoja de datos del fabricante.

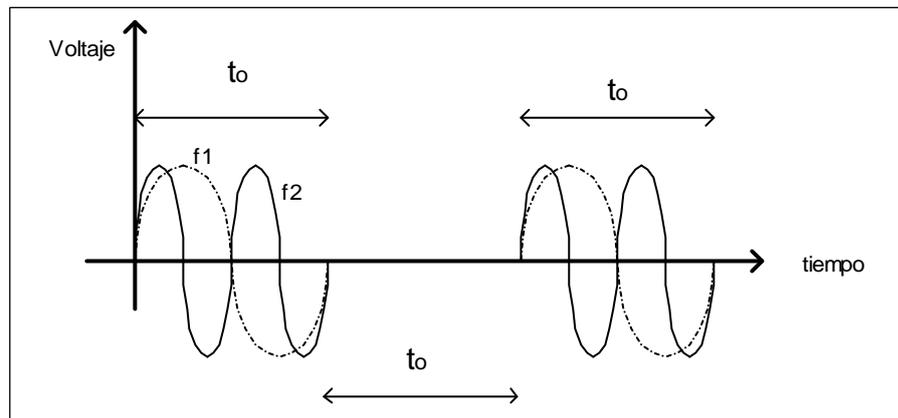


Figura 7.4: Diagrama de tonos DTMF

Esta característica de tiempos con tonos activos y tiempos sin tono nos impone otra limitación en la capacidad de transmisión de la información, en este caso en cuanto a la velocidad.

Para la recepción se trabaja con el circuito decodificador o receptor DTMF Mitel 8870, este recibe un par de tonos válidos y entonces lo decodifica en 4 bits en código binario los cuales se reflejan en el puerto del decodificador incluido en el circuito receptor y al mismo tiempo activa la salida ESt (Ver figura 7.5) que indica que en el buffer de salida del circuito hay presente un dato válido.

Con esta información, es posible que mediante un programa se pueda sondear esta salida del circuito decodificador usando el puerto paralelo de un computador; y dependiendo de su estado se leería la información del buffer del circuito, de esta manera habríamos logrado transmitir un nibble de información mediante radios VHF-AM usando los tonos DTMF.

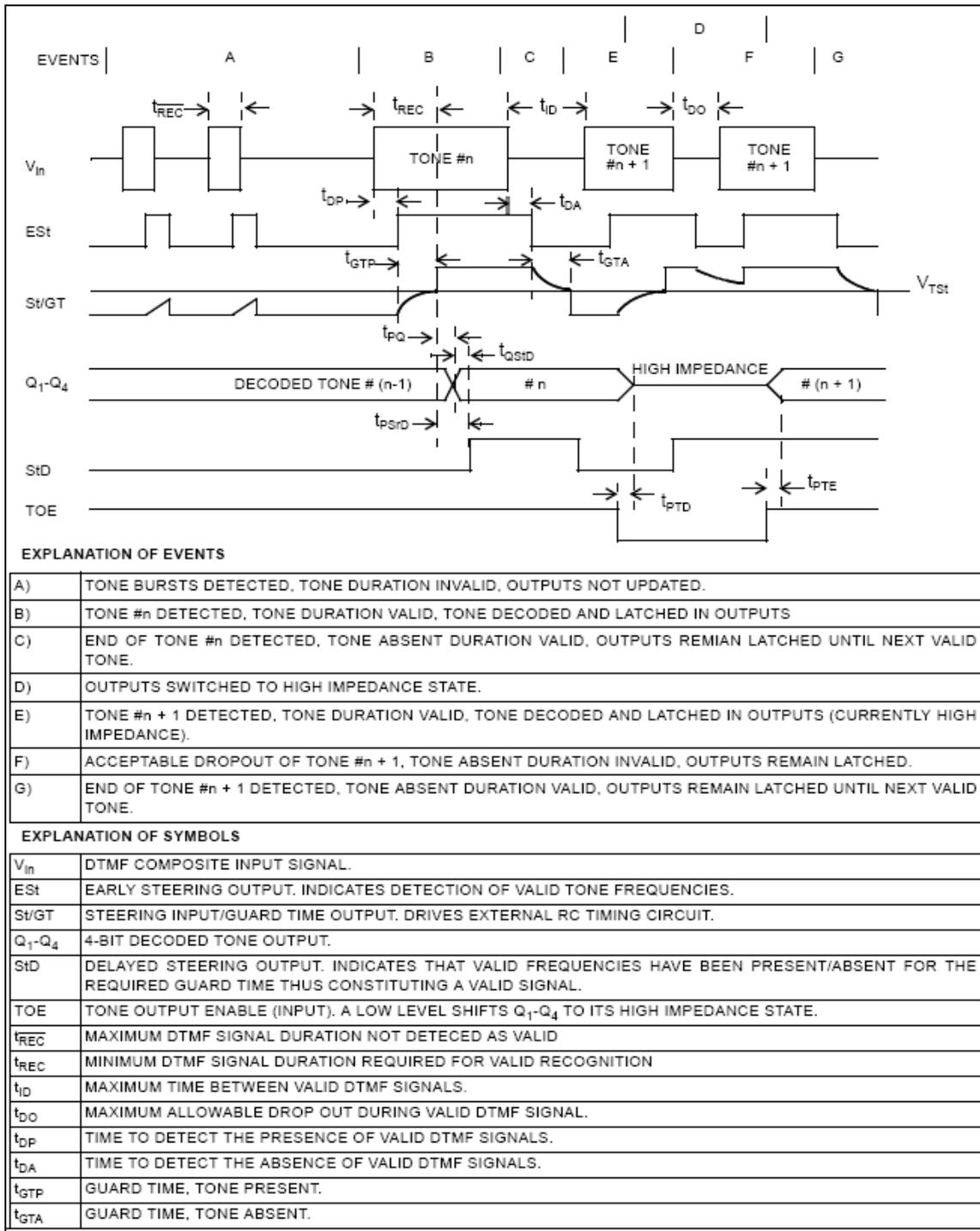


Figura 7.5: Diagrama de tiempos de Receptor 8870

7.4.2 Programas en Turbo C/C++

Se desarrolló un programa en Turbo C/C++ cuya función principal es poner en el puerto paralelo de la computadora *LPT1* información válida para el Generador DTMF 5089 y este pueda generar tonos DTMF según los valores fijados en el puerto paralelo.

Se confeccionó una tabla de correspondencia con los valores aceptados por los circuitos integrados generadores y receptores DTMF según la hoja de datos, de tal manera que sean estos valores los que el programa enviará al puerto paralelo con el tiempo especificado de presencia de tono y ausencia de tono. (Ver Tabla 7.1).

Mediante un algoritmo se hizo que el programa recibiera por teclado los valores aceptables que se quisiera enviar, en el programa se creó en primer lugar rutinas para el manejo del UART, tanto para recibir información como para transmitirla por el puerto serial. La información que se desea transmitir es la posición de un punto en un plano cartesiano y es obtenida de un equipo GPS en forma serial.

Tal como se describe anteriormente el protocolo que se usa es el NMEA 0183 y se escogió la sentencia GPGGA que el equipo GPS nos envía. Esta sentencia tiene el siguiente formato:

\$GPGGA,011048,1451.242,S,07457.563,W,1,04,1.8,579.9,M,-19.3,M,*

En las primeras seis posiciones siempre se recibe el encabezado \$GPGGA, luego sigue una coma que delimita de los siguientes seis dígitos que nos dan

información de la hora UTC, luego sigue otra coma que delimita de los siguientes ocho valores que nos dan información de la latitud en decimales, luego otra coma seguido de un indicador de latitud N (si es latitud Norte) y S (si es latitud Sur), sigue otra coma, luego continúan nueve dígitos que nos dan información de la longitud en decimales y también delimitado por comas un indicador de longitud E (si es longitud Este) y W (si es longitud Oeste). El signo \$ nos sirve como identificador de inicio de trama y el signo * como fin de trama.

Esta sentencia se recibe en forma serial y se almacena en un arreglo en memoria con las mismas posiciones con que son recibidas de la siguiente manera:

\$	G	P	G	G	A	,	0	1	1	1	4	9	,	1	4	5	1
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
.	2	4	4	,	S	,	0	7	4	5	7	.	5	6	6	,	W
18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35

Luego se procede a hacer una verificación simple por posiciones, es decir, verificar que en la posición 0 esta el valor \$, en las posiciones 1 al 5 esta el encabezado GPGGA, que en la posición 23 este el valor S y en la posición 35 este el valor W.

Luego estos valores de Latitud y Longitud son convertidos a minutos. A estos valores se les resta los valores de Latitud y Longitud de Referencia respectivamente obteniendo como resultado un diferencial de longitud y un diferencial de latitud denominados como Registros Delta de Latitud (S_DeltaReg) y Registro Delta de Longitud (W_DeltaReg).

Una vez que se tiene estos valores actualizados se procede a ensamblar la ADS Frame según el algoritmo descrito.

7.4.3 Primera Prueba.- Funcionamiento del protocolo

En las primeras pruebas la Transmisión y Recepción, el envío de información hacia el circuito generador DTMF se hizo por el puerto paralelo (LPT1) de la computadora. El Decodificador DTMF 8870, decodifica con 4 bits entonces se recibe dichos datos por el puerto de estado de LPT1 (*status port*).

7.4.3.1 Resultados

En esta primera prueba se puede demostrar que es posible transmitir tonos duales multi-frecuencia a través de radios VHF – AM y recuperar la información transmitida en la recepción, en resumen se logra recibir la tabla transmitida, pero con algunos problemas de sincronismo.

7.4.3.2 Análisis de resultado

Para diferentes valores de tiempos de presencia/ausencia de tonos se reciben en algunos casos la tabla completa, luego la siguiente transmisión con pérdidas de 2 a 3 caracteres por tabla (tabla = 16 caracteres). (*Ver Tabla 7.1*)

Por ejemplo para: PresenciaTono = 500 milisegundos y AusenciaTono = 100 milisegundos obtenemos trama completa con pérdidas de los caracteres 1,2,3

Para Presencia Tono = 300 milisegundos AusenciaTono = 100 milisegundos recibimos la primera trama completa con pérdidas sucesivas de caracteres en las siguientes transmisiones.

7.4.3.3 Primera modificación y resultados

Analizando detenidamente el diagrama de tiempos del 8870 se determina que en la recepción el tiempo de presencia de tono se debe disminuir en 20 milisegundos para compensar la falta de sincronismo.

Se modifico el programa con los valores anteriores y se creo por software un buffer en la recepción de los datos, con la finalidad de almacenar los 16 caracteres recibidos y luego mostrarlos.

A continuación se muestra una captura de la pantalla del programa de recepción:

```

RECEPCION

Ingrese tiempo de presencia de tono (ms): 150

Ingresetiempo de sincronismo (typ=20ms): 20

          ESTADO: NO HAY DATO RECIBIDO!!!      X: Dato errado!!!

          34567890*##ABCD1234567890*##ABCD1234567890*##ABBCD123456
7890*##ABCD1234567890*##ABCD1234567890*##ABCD1234567890*##ABCD1234
5667890*##ABCD1234567890*##ABCD1234567890*##ABCD1234567890*##ABCD1
2234567890*##ABCD1234567890*##ABCD1234566788900*##ABCD1234567890*
1234567890*##ABCD##ABCD
  
```

De 268 caracteres transmitidos se repitieron 5:

- En la primera fila se repitieron el caracter # y el B
- En la tercera fila se repitieron el caracter 6
- En la cuarta fila se repitieron el caracter 2, 6,8 y 0

Se observa que ya no existen caracteres perdidos pero si se tiene repetición de caracteres.

7.4.3.4 Segunda modificación y resultados

Teniendo como base estos resultados se modificó del algoritmo de recepción; se incorporó una rutina de verificación de *glitches* (ver glosario) con un

contador programable, para no leer dos veces un carácter. Esto se tiene que hacer porque el estado lógico de un pin se puede alterar debido al ruido eléctrico presente en los circuitos, esto probablemente era la causa de la repetición de los caracteres.

Por software se crea un buffer de Recepción con los datos que se van a Transmitir para controlar que la trama llegue sin errores, es decir se implementa por software un primitivo sistema de verificación de la trama recibida por comparación con la trama esperada.

Luego de estos cambios, se probó transmitir nuevamente y se logra recibir la trama completa sin errores, directamente de generador a decodificador (en banda base), sin pasar por las radios AM.

7.4.4. Segunda Prueba.- PTT automático

El siguiente paso que se tomo fue hacer que el Key o PTT (*Press To Transmit* o *Push To Talk*) sea automático, con una de las salidas de control de LPT1.

Para que la señal pueda ser transmitida, es necesario poner a potencial cero (tierra) el pin PTT de la radio. Para esto se uso una resistencia de $2000\ \Omega$ y un transistor **2N2219** en configuración de colector abierto.

Este circuito funciona sin problemas, pero durante la transmisión de información se observa que es necesario un determinado tiempo de establecimiento del PTT, es decir un tiempo de espera antes de modular las radios con los tonos dtmf, hasta que la portadora de RF pueda estabilizarse y llegue aprox. al 90% de su potencia de salida. El tiempo de respuesta de conmutación del transistor es despreciable respecto de los valores en el rango

de milisegundos del tiempo de establecimiento. Este tiempo se hizo programable por software y experimentalmente encontramos en un inicio que el tiempo idóneo es de 400 a 500 milisegundos de PTT antes de modular la señal con los tonos DTMF.

Se hicieron pruebas de esta manera y se observa que hay algunos tonos (1,4,5,a,b) que se pierden cuando se transmiten solos, es decir un solo carácter por trama.

Esta pérdida de información se tuvo que analizar en el osciloscopio, en la salida del generador de tonos DTMF encontramos lo siguiente:

Cuando no se transmite tono hay un nivel DC de aproximadamente 1 Voltio en la salida del Transceptor DTMF, pero cuando a la salida se presenta el tono, este nivel DC aumenta a 2.3 Voltios más el nivel de voltaje de la señal AC (Ver figura 7.6)

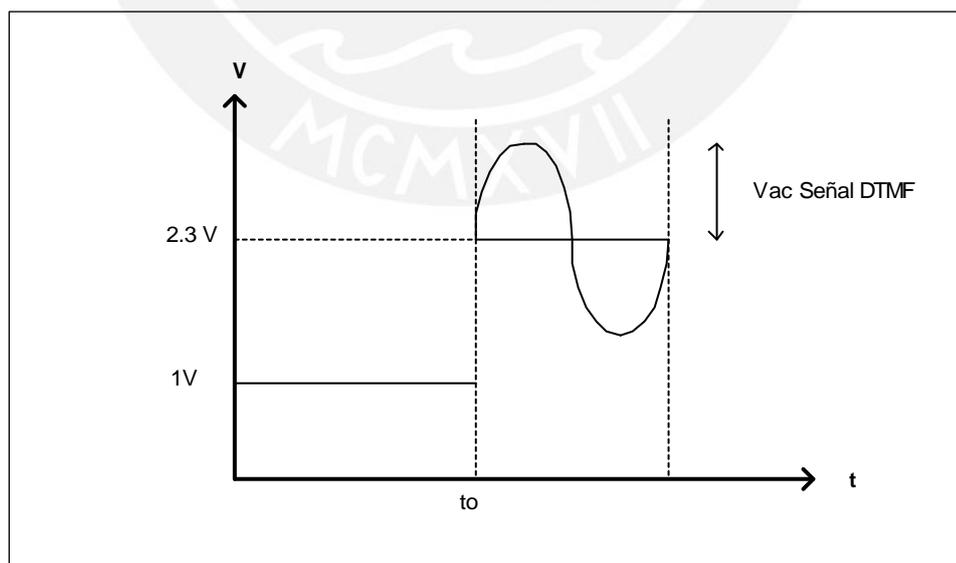


Figura 7.6: Análisis de salida DTMF

El tiempo T_o indica el momento en que se empieza a generar el tono, esto origina un escalón a la entrada de la radio, lo que inicialmente lo satura, y ocasiona que el tono inicial no se transmita.

Para solucionar este problema se colocó un condensador de 0.1 microfaradios a la entrada de la radio y se modificó la Transmisión, enviamos un tono simple (*single tone*) por delante de cada trama, para estabilizar la radio antes de enviar los tonos de información.

El *single tone* es una opción del generador DTMF, mediante el cual se genera un tono de una cierta frecuencia (no dual) que el decodificador no entiende, es decir no decodifica ningún valor, de esta manera la recepción no se ve afectada con una posible falsa información.

Se selecciona enviar un tono de 1477 Hercios por un tiempo determinado, se empezó bajando el tiempo de Transmisión del tono desde 1000 milisegundos hasta 200 milisegundos, con este valor se logra transmitir normalmente sin pérdidas de caracteres, y además la primera trama llega sin errores.

- Haciendo pruebas posteriores se consigue recepcionar los datos sin problemas de pérdidas de caracteres con los siguientes valores:

Tiempo de PTT = 100ms

Tiempo de Single Tone = 200ms.

Una vez que se encontró los valores adecuados para los que las pérdidas de información se reducen al mínimo, se procedió a confeccionar una tarjeta de prueba.

Las pruebas en el laboratorio se hicieron usando atenuadores entre las radios para evitar interferir con el funcionamiento normal de la banda aeronáutica.

La Potencia de los radios es de 8W que equivale a 39 dBm y los receptores tienen una sensibilidad de 0.5 micro voltios que equivale a -66dBm.

Se debe poner un atenuador mayor o igual a 105 decibeles. Ver figura 7.7

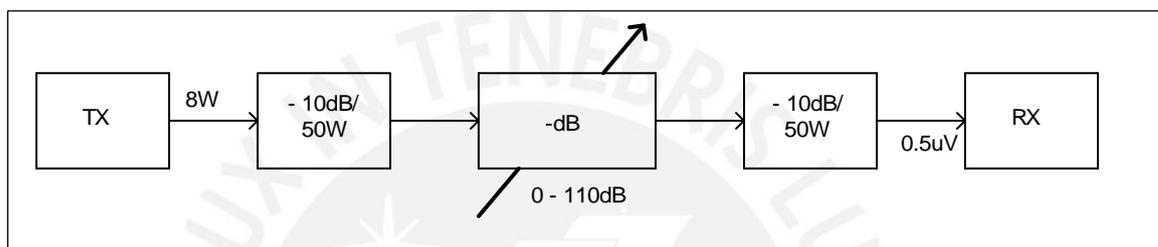


Figura 7.7: Conexión de atenuadores

7.4.5 Tercera Prueba.- Cobertura y factores externos

Se hicieron pruebas instalando la tarjeta en una movilidad para comprobar la cobertura y la pérdida de la información por factores externos mas reales como pueden ser: interferencia, obstáculos (línea de vista), desvanecimiento, cobertura de la señal por la potencia de los radios transceptores, la ubicación de las antenas.

7.4.5.1 Resultados

- Se hicieron pruebas en el laboratorio con todos los equipos: Receptor GPS, Radio Transmisor, Radio Receptor, computador transmisor y computador receptor, tal como en la figura 7.8.

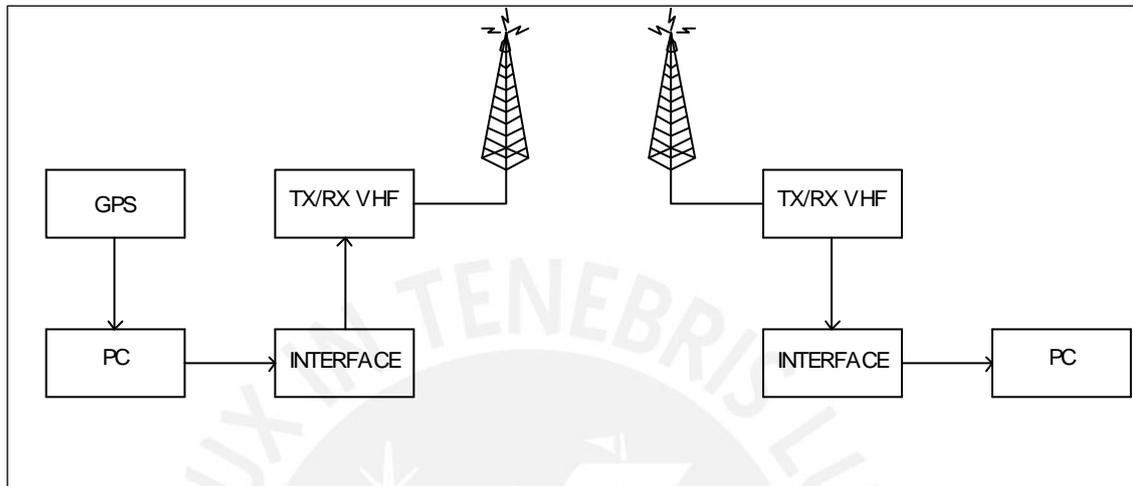


Figura 7.8: Diagrama de bloques del sistema

El redondeo de los datos y la cantidad de bits nos permite una precisión de un decimal, esta precisión es baja, dada la precisión que se necesita para esta aplicación. Como parte del algoritmo, el programa escribe los datos decodificados en un archivo en el disco duro cada vez que recibe una trama.

Se verificó que en lugares donde hay línea de vista, la información llega sin errores. La movilidad procedió a hacer un recorrido por toda la pista de aterrizaje del Aeropuerto Int. Jorge Chávez, de cabecera Sur a cabecera Norte captando la información sin errores.

Se hicieron pruebas con la movilidad equipada con los equipos transmisores y con un software de visualización del móvil (software desarrollado por Corpac) con los datos decodificados y se pudo observar saltos de más de 100 metros en el programa de visualización.

7.4.5.2 Modificaciones y resultados

Con la finalidad de disminuir el tamaño de los saltos, se procedió a aumentar los campos de Latitud y Longitud de 10 a 12 bits y además se aumentó un campo de verificación de la trama (*checksum*), y se envía un carácter de inicio(*SOF*) y otro de fin de trama(*EOF*) con lo que la trama queda de la siguiente manera:

#	Id
Δlat	Δlat
Δlat	Δlon
Δlon	Δlon
Chk	*

Se volvieron a hacer pruebas con la movilidad equipada con los dispositivos de transmisión a lo largo de la pista de aterrizaje, con estos nuevos valores se logra disminuir la magnitud de los saltos al orden de 20 a 30 metros.

7.4.6 Análisis de errores del sistema

Sobre los saltos habría que decir lo siguiente:

Un primer error se produce en el equipo GPS por el principio de funcionamiento de este equipo, con errores admisibles para el uso civil de un promedio de 100 metros en ese momento. Actualmente este error ha sido reducido por el Gobierno de los Estados Unidos a aproximadamente de 5 a 20 metros, ya que se eliminó la opción de *Select Availability* (disponible solo para uso militar), que

introducía un ruido a la señal proveniente de los satélites con la finalidad de disminuir la precisión de las medidas.

Un segundo error se introduce por la cuantización de los datos recibidos, ya que el tamaño del campo (12 bits) nos obliga a hacer redondeos en el programa de transmisión.

Un tercer error se produce durante el procesamiento de los datos en el programa de Recepción.

7.4.7 Cuarta Prueba.- Pruebas de campo en Nasca

Hechas las pruebas en Lima, y teniendo conocimiento de los factores que afectan la precisión de los datos enviados, se realizó un viaje a la ciudad de Nasca para realizar pruebas de cobertura y de precisión reales, esto significa, con todo el equipamiento de pruebas instalados en aeronaves. Se hace también un reconocimiento de los equipos de comunicaciones que existen a bordo de las aeronaves, para su posible uso en este sistema.

En Nasca se instalaron los equipos: radios, antenas, baterías y las tarjetas DTMF. Luego se procede a realizar una prueba de cobertura en la móvil a lo largo de la carretera Panamericana con dirección a las *Líneas de Nasca*, que es la zona donde se quiere tener cobertura tanto de ondas de radio como del programa de visualización.

7.4.7.1 Observaciones

- Se detecta que en cierto punto hay cambio de grados en las coordenadas geográficas (de $74^{\circ} 99'$ a $75^{\circ}00'$), lo que en el programa no estaba previsto,

por lo que se procedió a la modificación del programa para tomar en cuenta este detalle.

- La cobertura de ondas de radio es buena en los lugares donde hay línea de vista. Sin embargo en los lugares donde existen obstáculos físicos (cerros), los tonos llegan incompletos, esto por la característica de transmisión en VHF.

7.4.7.2 *Modificaciones en sitio*

Se decide aumentar la precisión de los datos, para lo cual se debe modificar la estructura del ADS FRAME, esto significa aumentar un byte más a la trama, lo que implica campos de 14 bits para la Latitud y la Longitud.

#	Id
Δlat	Δlat
Δlat	$\Delta\text{lat}/\Delta\text{lon}$
Δlon	Δlon
Δlon	Chk
*	

Se modifica el algoritmo para obtener el factor que se multiplica al diferencial de posición para obtener el valor de delta de posición, de la siguiente manera:

$$2^n / D_{\text{millas}} = \text{factor}$$

En donde tenemos que $n=14$ bits y $D=32$ millas de área de cobertura, se obtiene el factor de 512.

El paso siguiente fue montar el equipo sobre la aeronave y esperar su despegue, para su observación en el monitor en tierra.

Con esta estructura se procedieron a hacer las pruebas en la movilidad, obteniendo las siguientes conclusiones:

1. Se pudo observar el movimiento de la aeronave en la pantalla de la estación base, con lo que se comprueba el buen funcionamiento del sistema diseñado.
2. Se observan aun ciertos saltos del orden de 5 a 10 metros, debido a los mismos factores mencionados anteriormente.

7.4.8 Análisis de rendimiento del sistema

Para el análisis de consumo de energía y ciclo de trabajo de los radios de comunicación, es necesario hacer los siguientes cálculos:

El circuito de vuelo de cada aeronave dura aproximadamente 30 minutos, de donde podemos deducir:

1 trama / 6 segundos ---> 300 tramas por vuelo.

$(1/6) \times 30 \text{ minutos} = 5 \text{ minutos de uso de la radio por vuelo.}$

Obtenemos **16.67 %** de ciclo de trabajo de las radios trabajando en estado ON/OFF.

Cuando se realizaron las pruebas de estos valores no se apreció recalentamiento apreciable de las radios de comunicación.

7.5 Segunda Etapa

En esta etapa se elimina el computador de a bordo y se reemplaza por un circuito que incluye un microcontrolador que maneje el programa de transmisión de datos, la recepción del GPS y las interfaces de entrada y salida.

Se hace un estudio del sistema de acceso de varias aeronaves a la estación base y su implementación.

7.5.1 Estudio del método de Acceso al medio

Este análisis trata del estudio de la forma de acceso de varias aeronaves a la estación base. En esta etapa se considera inicialmente la fabricación de 3 equipos, que es algo que nos facilita de cierta manera el acceso. Pero en el caso de que entren en funcionamiento los 16 previstos tendremos que analizar el problema en la actualización de la información dado que cada trama dura aproximadamente 1.4 segundos:

Son 11 tonos para enviar, cada uno de 100 milisegundos (50 de presencia de tono y 50 de ausencia de tono), se tiene 1100 milisegundos. Además de los 100 milisegundos de PTT y 200 milisegundos de tiempo de tono simple, entonces tenemos 1400 milisegundos.

Los principales tipos de acceso al medio analizados son el de Acceso Múltiple por División de Tiempo (TDMA) y Acceso Múltiple por Detección de portadora (CSMA).

A continuación se analizan diferentes casos que se pueden implementar para dar solución al esquema presentado en este trabajo:

7.5.1.1 Caso TDMA que sincroniza con reloj transmitido de tierra.

En esta situación la estación base transmitiría una trama de sincronización hacia todas las estaciones la cual indicaría que ese es el *time slot 0* o tiempo "cero" y que a partir de ese instante cada equipo debería sincronizarse con el temporizador interno de su microcontrolador, esto significa que en ese momento empezaría el sistema de acceso al medio TDMA.

Este reloj o señal de sincronización podría ser transmitido periódicamente, por ejemplo cada 60 segundos hacia las estaciones móviles de tal forma que se pueda sincronizar todos los relojes de los equipos cada cierto tiempo y así corregir posibles fallas en el sincronismo, así tampoco dependeríamos totalmente de la hora del GPS ya que llegaría a ocurrir que las estaciones comiencen a perder sincronización y podrían a empezar a superponerse los *timeslots* y llegarían a existir dos transmisiones simultáneas produciendo colisiones indeseadas y no programadas en este sistema TDMA.

Como una variación se podría diseñar este caso también de tal forma que luego de enviada la trama de sincronización, al siguiente segundo igual a cero sería el *timeslot* cero de inicio del algoritmo TDMA.

7.5.1.2 Caso CSMA

El diseño del circuito electrónico tiene previsto una entrada digital TTL que se puede usar para poder sondear la presencia de portadoras en el medio y de acuerdo a ese escenario decidir la transmisión de su información. Se tendría que implementar por software determinado tipo de algoritmo CSMA que sea capaz de manejar este tipo de situaciones. La detección se haría usando el pin

de detección de portadora (CD) que poseen los radios receptores VHF-AM el cual se conectaría a la entrada del microcontrolador, esta entrada estaría polarizada como colector abierto. La salida CD de los radios receptores VHF AM es un relay que se cierra cuando el receptor ha detectado una portadora en la frecuencia en que este esta trabajando.

Este caso podría usarse en aquellas situaciones en que tenemos muchas estaciones y el sistema TDMA colapse debido al tiempo de duración de la trama.

7.5.1.3 Caso de Transmisión de Posición por petición

En este caso, el algoritmo consistiría en que la estación base tendría que enviar una trama de petición de información a cada aeronave para que transmita su posición. Este procedimiento se puede realizar de dos formas:

- En forma periódica, en este caso se programaría un tiempo determinado en el que la estación base estaría solicitando la información de posición a una aeronave especifica o a un grupo de ellas.
- Por eventos, en el caso de la estación base necesite información de determinada aeronave en el momento que lo desee.

7.5.1.4 Caso TDMA que sincroniza con hora UTC del GPS

En este caso el algoritmo de acceso fija su *timeslot* cero o tiempo de inicio del procedimiento TDMA usando el reloj recibido del GPS. El principal problema de este caso ocurre con la exactitud y retardo del reloj GPS y la recepción serial de este reloj.

Analizando la salida de datos del GPS, aunque teóricamente se configure la Actualización de datos del receptor GPS en un tiempo determinado, que es el tiempo en el que el GPS envía una trama de información por el puerto serial, esto no ocurre en la realidad, por ejemplo si se configura en el receptor GPS para que envíe trama de información por el puerto serial cada segundo esto no ocurre con exactitud y mas bien se da cada 2,3 y en algunos casos hasta 4 segundos como máximo de forma variable. Esto definitivamente complica la forma de acceso. Teniendo un primer y fácil raciocinio nos conduce a tener que hacer cada *timeslot* de 5 segundos. Conociendo este problema, sincronizar con los datos del GPS se torna complicado.

Como una variación de este algoritmo se tiene la siguiente opción:

Que el equipo móvil posea un reloj interno que se va actualizando cada vez que recibe información del GPS, sin que sea necesario recibir una trama GPS para enviar la información en el momento que le corresponda. Es decir con una primera trama recibida del GPS sincronizan el reloj para el inicio del procedimiento TDMA y luego este reloj interno reconvierte en el maestro que determina las transmisiones, pero que se actualiza con el reloj del GPS. Esto implica que si su reloj interno le indica que le toca transmitir y aun no ha recibido una nueva trama GPS, el equipo transmite su posición anterior.

7.5.1.5 Opción elegida

Se escoge la opción: Caso TDMA que sincroniza con hora UTC del GPS.

El procedimiento sería el siguiente: Fijamos un *timeslot* de 2 segundos para cada aeronave, como son tres aeronaves entonces tenemos un periodo de 6

segundos. Lo que significa que cada aeronave transmitirá su posición cada 6 segundos.

El temporizador interno interrumpe el programa cada 5 milisegundos, lo que nos indica que en la interrupción 200 habrá transcurrido 1 segundo, aquí se actualiza los segundos. La lectura del USART también se hace por interrupciones, se asigna un espacio de memoria para que de acuerdo a un puntero de escritura se almacenen los datos. Cuando ha terminado de recibir una nueva trama GPS, ésta se verifica y se actualiza la información en la memoria RAM, los datos que se actualizan son los de Latitud, Longitud y la hora UTC, luego se activa una bandera que indica que se ha recibido una trama completa para empezar el chequeo y la Actualización de los datos.

Esta opción es la que se desarrollo finalmente en los algoritmos de los programas de transmisión en el PIC y en la Recepción en la PC en instalada en la Estación Base.

7.5.2 Diseño del Circuito Electrónico

En esta etapa se elimina la PC de a bordo y se reemplaza por un microcontrolador, tal como se aprecia en la figura 7.9:

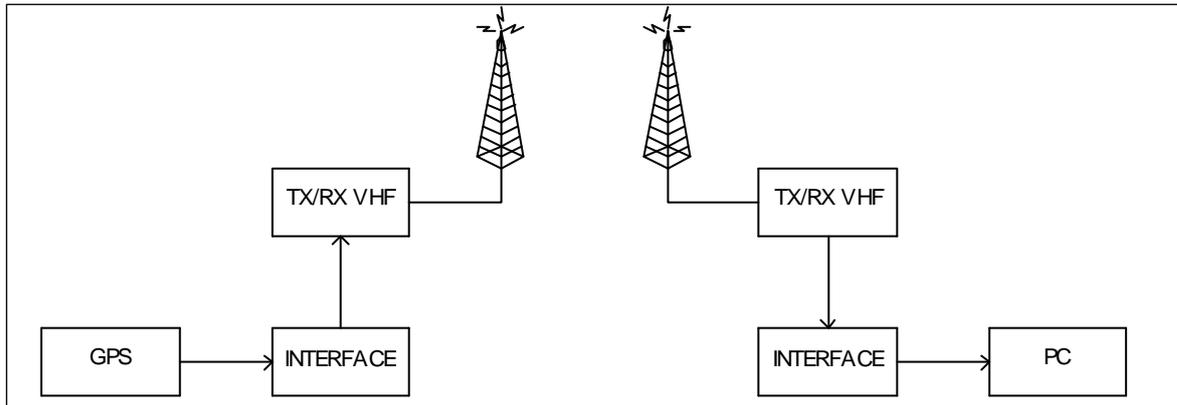


Figura 7.9: Diagrama de bloques del sistema modificado

7.5.2.1 Estudio del microcontrolador

Analizando los diferentes tipos de microcontroladores que existen en el mercado, se decidió por el **PIC16F877** de **MICROCHIP**, en primer lugar porque los microcontroladores PIC son de bajo costo y existen en el mercado y este en particular por el tamaño de memoria que posee.

Principales características del Microcontrolador PIC16F877

- Arquitectura RISC, de pocas instrucciones.
- 8K x 14 bytes de memoria Flash EEPROM de programa, lo que permite un borrado mas rápido de la memoria para su reprogramación.
- 368 x 8 bytes de Memoria RAM de datos
- 256 x 8 bytes de memoria EEPROM de datos
- Por tener la memoria Flash, se puede depurar o programar *in-circuit*, es decir hay más facilidad para la depuración del programa, lo que permite la detección y corrección de errores con mayor facilidad.

- Modos de direccionamiento de memoria: directo, indirecto y relativo.
- Capacidad de 14 Interrupciones
- 5 puertos de entrada/salida (A, B, C, D y E)
- Puerto de comunicación serial (USART)
- 03 Temporizadores y/o contadores
- Watch Dog Timer
- Juego de 35 instrucciones
- Conversor Analógico Digital de 10 bits

7.5.2.2 Estudio del Transceiver DTMF M8888

Para esta etapa se consideró el uso de un solo circuito integrado que haga de transmisor y receptor, para esto se seleccionó el Transmisor/Receptor de tonos DTMF Mitel M8888. La etapa de recepción de este circuito esta basado en el receptor usado en la primera etapa que es el Mitel 8870.

La característica principal de este circuito integrado es que tiene la posibilidad de ser programado para seleccionar el modo de funcionamiento, ya sea de receptor o de transmisor (ver Tabla 7.2), posee registros de programación que permiten además seleccionar la modalidad BURST (50milisegundos/50milisegundos de presencia – ausencia de tono) automático y Single Tone (ver Tabla 7.4)

Mediante un microcontrolador o PC se puede tener acceso al registro de estado, a los dos registros de control y dos registros de datos (ver Tablas 7.2, 7.3, 7.4 y 7.5).

La tabla de codificación y decodificación de tonos es la misma que los circuitos 8870 y 5089 dado que son valores estándar.

7.5.2.3 Circuito de Generación y decodificación de tonos DTMF

En esta etapa se usa el circuito integrado Mitel 8888. En la salida de tono (ver Tabla 7.3), se coloca un potenciómetro R15 que servirá como control del nivel de salida del tono DTMF, luego pasa por una etapa de desacoplo DC usando un condensador cerámico C6 de 0.1 microfaradios y luego pasa a un circuito seguidor de señal que sirve como adaptador de impedancia entre la salida de tono DTMF y la entrada de audio (MIC) del radio, para esto se usa el amplificador operacional LM358 U7.

De igual manera en la entrada de tono al circuito DTMF se coloca un circuito amplificador de ganancia unitaria para adaptar la salida de audio del radio a la entrada de señal del circuito DTMF; igualmente se usa el LM358, se incluye además un circuito de atenuación de la señal usando el OPAMP interno del DTMF. Cuando el circuito 8888 detecta un par de tonos válidos, la salida EST se activa, normalmente esta salida esta en estado inactivo.

Este circuito DTMF M8888 trabaja con un cristal de 3.58 MHz.

Para la programación de los registros internos del circuito se debe tener en cuenta los siguientes valores:

Tabla 7.2: Registros del 8888

RSO	RD	WR	Función
0	1	0	Escribir en el transmisor
0	0	1	Leer del receptor
1	1	0	Escribir en el registro de control
1	0	1	Leer del registro de estado

Tabla 7.3: Descripción del Registro de Control A del 8888

Bit	Nombre	Función	Descripción
B0	TOUT	Salida de tono	Un 1 lógico habilita la salida del tono
B1	CP/DTMF	Modo de Control	En modo DTMF (0 lógico) el dispositivo es capaz de recibir y generar tonos DTMF. En 1 lógico se habilita la función de detección de progreso de llamadas.
B2	IRQ	Habilita Interrupción	Un 1 lógico habilita el modo de interrupción.
B3	RSEL	Selecciona registro	Un 1 lógico selecciona el registro de control B en el siguiente ciclo de escritura, luego de lo cual automáticamente regresa al registro A

Tabla 7.4: Descripción del Registro de Control B del 8888

Bit	Nombre	Función	Descripción
B0	BURST	Modo Ráfaga o Bursa	Un 0 lógico habilita el modo Burst con los valores DTMF fijados y con duración fija de 50 ms de ausencia y presencia de tono. Un 1 lógico permite cambiar la duración de los tonos
B1	TEST	Modo Test	Un 1 lógico habilita esta opción.
B2	S/D	Generación de tono simple o dual	Un 0 lógico permite que señales DTMF sean producidas. Un 1 lógico permitirá generar tonos simples dependiendo del valor de B3
B3	C/R	Tonos en columnas/filas	Usado conjuntamente con en B2, un 0 lógico seleccionara las filas.

Tabla 7.5: Descripción del Registro de Estado del 8888

Bit	Nombre	1 lógico	0 lógico
B0	IRQ	Ocurrió una interrupción B1 o B2 esta activo	Interrupción inactiva. Se desactiva después que se lee el registro
B1	Reg. Transmisión de datos vacío (modo Burst)	La duración de la pausa termino y esta listo para mas datos	Si es que el registro de estado ha sido leído o si no esta en modo Bursa
B2	Reg. De Recepción datos lleno	Dato Valido en el registro de datos	Registro de estado fue leído
B3	Delayed Steering	En ausencia de tonos DTMF	Si hubo detección de un par de tonos validos

Un parámetro eléctrico importante que se debe mencionar y que se analizó anteriormente es el valor de 2.5 DC presente en la salida de tono cuando no hay tono presente.

El circuito microcontrolador se polarizó tal como la hoja de datos del fabricante lo sugiere, se usó un reloj de 20 Megahertz.

Igualmente se polarizó la entrada MCLR para reinicializar el dispositivo mediante un pulsador, se reinicia cuando se pone a potencial cero esta entrada, normalmente esta polarizada con voltaje y una resistencia R13 de 4.7K.

7.5.2.4 Circuito de Interfase para Entradas y Salidas

El puerto paralelo número 7 del microcontrolador se usa como entradas de la siguiente manera:

- RB4, RB5, RB6 y RB7 son las entradas que proporcionan información de la identificación del equipo, se selecciona mediante un *dipswitch*, se prevé un máximo de 16 aeronaves.
- RB3 es una entrada con un pulsador que indicara al equipo que debe enviar de una trama de prueba incondicionalmente.

Para la opción de PTT automático, se incluye un circuito con el transistor T3 2219 polarizado de forma universal con las resistencias R11 y R12, que será cortado o saturado por el PIC usando el puerto RB0, para generar la tierra necesaria de manera que el radio pueda modular la señal que entrada.

Adicionalmente en el panel frontal se incluye un LED que será manejado por el PIC en el puerto RB1 para indicar cada vez que se envía un dato, cortando o saturando el transistor T2 2N3904 polarizado con las resistencias R9 y R10.

Se incluye en RB2 una interfase para lectura de Squelch, con un transistor PNP T1 2N2219 en polarización universal con las resistencias R6, R7 y R8, teniendo la entrada en la base, la que se encontrara abierto o cerrado, esta entrada ocasionara que el transistor se corte o sature. Se incluye además un diodo N1 1N4148 que impide que la corriente circulen sentido contrario.

7.5.2.5 Circuito de Transmisión y Recepción Serial

Se incluye en el circuito un dispositivo conversor RS232/TTL para adaptar las salidas con niveles según el estándar RS232 del equipo GPS con el nivel TTL que maneja el USART del PIC16F877. Se considera en el circuito un puente JP1 para el caso de que algunos equipos GPS manejen en su salida niveles TTL. El circuito usado es el MAX232.

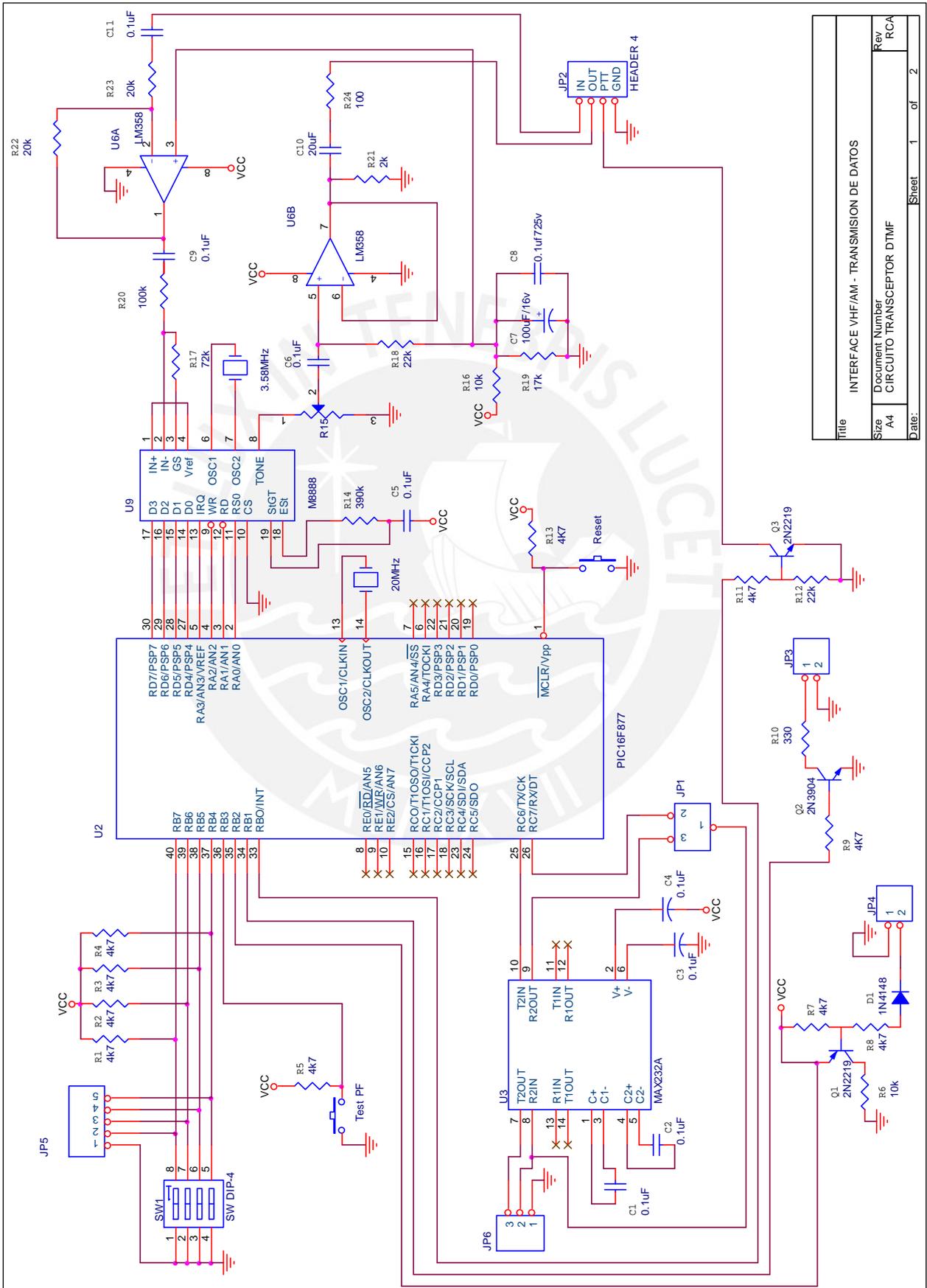
7.5.2.6 Análisis de parámetros eléctricos

Los siguientes datos tabulados se recogieron de las hojas de datos del fabricante para los circuitos integrados, así como para los transistores las corrientes de saturación y corte.

Tabla 7.6: Consumo de corriente de tarjeta

I_{pic}	250.0	mA
I_{dtmf}	11.5	mA
I_{2N2219}	10.0	mA
I_{2N3904}	15.0	mA
I_{test}	1.0	mA
I_{max232}	10.0	mA
I_{dipswitch}	4.0	mA
I_{lm358}	50.0	mA
I_{ref}	1.0	mA
I_{2N3906}	10.0	mA
I_{total}	362.5	mA

7.5.2.7 Diagrama Esquemático



Title		INTERFACE VHF/AM - TRANSMISION DE DATOS	
Size	Document Number	Rev	RCA
A4	CIRCUITO TRANSCREPTOR DTMF		
Date:	Sheet	1	of 2

7.5.3 Software

El software usado es de propiedad de CORPAC, tanto el Turbo C/C++ así como el MPLAB-ICD de Microchip.

7.5.3.1 Programa en Lenguaje Ensamblador

Una copia del código del programa y el diagrama de flujo se adjunta en los anexos.

En el programa en lenguaje ensamblador se mantiene las principales rutinas creadas en el programa en Lenguaje Turbo C/C++ para la transmisión en la primera etapa, salvo algunas características adicionales propias de la programación en lenguaje de bajo nivel que analizamos a continuación.

Se asigna un espacio en memoria fijo para almacenar los datos recibidos del equipo GPS, para el ensamble de la trama ADS.

En lenguaje ensamblador a diferencia de Turbo C/C++ no se pueden manejar valores reales con signo.

Como la Trama ADS es un protocolo orientado a bit redefinimos la trama de la siguiente manera para poder manejarlo en lenguaje ensamblador:

1	2	3		4	5	6			7	8	9	10	11
#	ID	+/-	LAT	LAT	LAT	LAT	+/-	LON	LON	LON	LON	CHK	*
0-3	4-7	8	9-11	12-15	16-19	20-21	22	23	24-27	28-31	32-35	36-39	40-43

En la primera fila tenemos los números de tono duales DTMF que se transmiten en la trama, que en total suman 11 tonos.

En la segunda fila tenemos la información que lleva cada tono transmitido, así tenemos que, el tono numero uno (1) es el que lleva los cuatro (4) bits de de

inicio de trama (STX) y el tono numero once (11) el que lleva los cuatro (4) bits de fin de trama (EOF).

En la tercera fila se indican los bits que corresponden a cada tono, además esta fila nos sirve para especificar en los casos de los tonos numero tres (3) y numero seis (6) en los que se comparte información.

En el tono numero tres (3) el bit numero ocho (8) es el que nos da el signo del valor que le sigue, que es el diferencial de Latitud.

En el tono numero (6) los bits numero veinte (20) y numero veintiuno (21) son los bits menos significativos del diferencial de Latitud; el bit numero veintidós (22) es el signo del diferencial de Longitud y el bit numero veintitrés (23) es el bit mas significativo del diferencial de Longitud.

Luego se hace una operación de OR entre todos los *nibbles* de información y el resultado se ensambla como el valor de chequeo de trama denominado *Checksum* en la trama ADS, que corresponde al tono diez (10).

7.5.3.2 Algoritmo de Retardo

Durante el programa se utiliza un procedimiento de retardo o *delay* la cual esta relacionado directamente con la frecuencia del oscilador del microcontrolador según las siguientes formulas:

Según la hoja de datos del PIC16F877 tenemos:

$$T_{cy} = 4 / F_{osc}$$

$$T_{delay} = [2 + (6 + n1) n2] T_{cy}$$

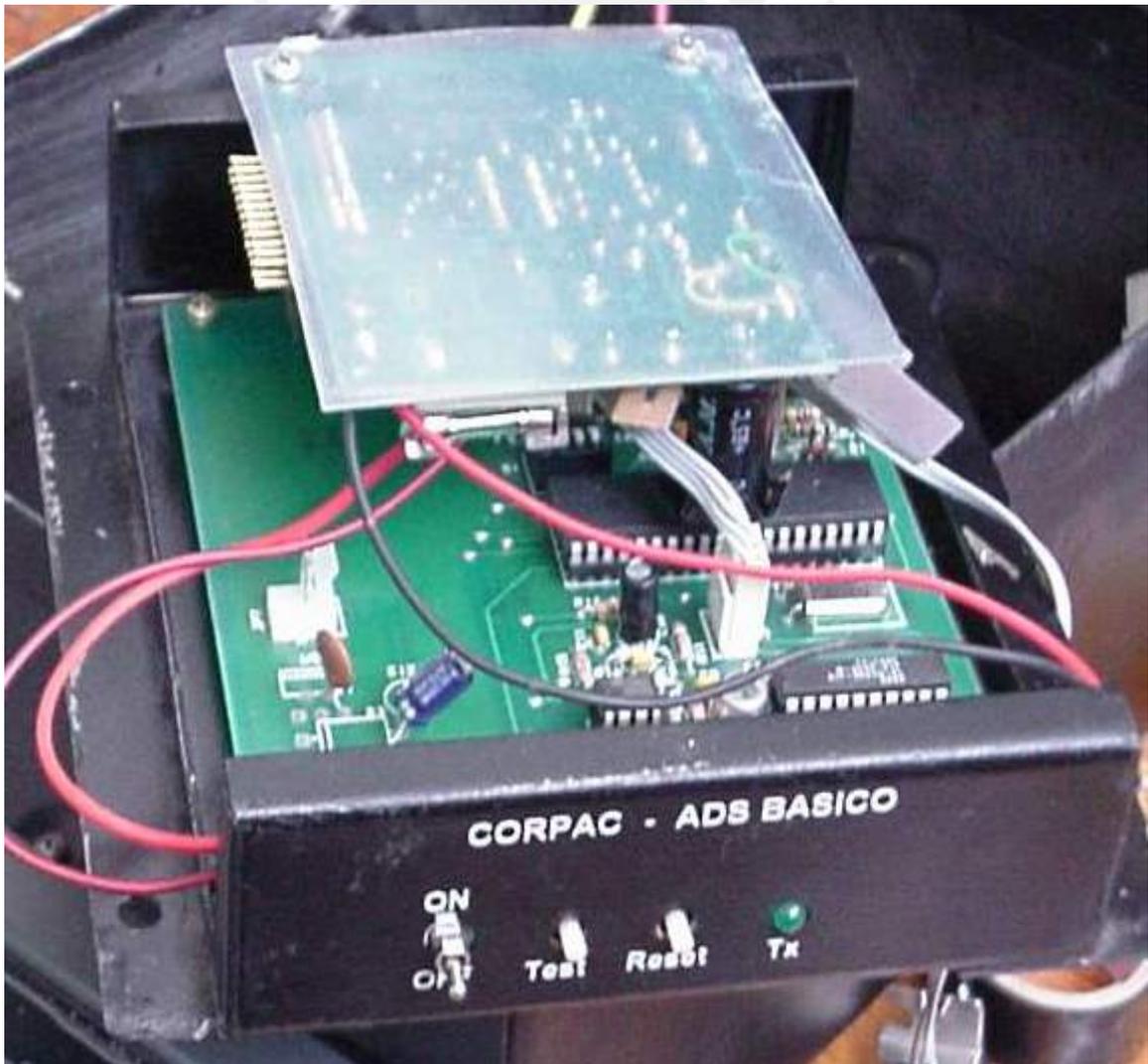
De donde obtenemos:

$$\text{Si } F_{osc} = 20 \text{ MHz entonces } T_{cy} = 0.2 \text{ } \mu\text{s}$$

Luego si queremos obtener un tiempo de retardo aproximado de 50 milisegundos entonces calculamos $n1=0xFF$ y $n2=0xFF$.

7.5.4 Alojamiento metálico del equipo

En la siguiente imagen se puede apreciar el prototipo terminado de equipo y que se uso en las pruebas finales. Aquí se puede apreciar el circuito inferior que incluye el microcontrolador y los periféricos, la tarjeta superior que esta invertida es la fuente de alimentación de 12 a 5 voltios.



7.5.5 Pruebas y Resultados

Se confeccionaron tres prototipos de equipos los cuales se instalaron para las pruebas, sobre tres distintas aeronaves que sobrevolaron la zona de las Líneas de Nasca en el departamento de Ica, Perú.

En el cuadro siguiente se aprecia el equipamiento instalado en cada una de las aeronaves:

Matricula Aeronave	OB-1001	OB-1334	OB-1192
Identificación	0	1	2
Equipos GPS	Garmin 55	GPS 90	Trimble Pro GPS
Equipo de Radio VHF	Becker	DX Radio System	DX Radio System
Energía	Con fuente externa de Radio	Con fuente externa de Radio	Con fuente externa de Radio
Baterías	Bosch 12 VDC	Bosch 12 VDC	Yuasa 12 VDC
Atenuadores	NO	10 dB/50 W	3 dB/7W
Cables de RF	Blindado RG223	Blindado RG223	Blindado RG223

Las tres aeronaves fueron contratadas por CORPAC para realizar vuelos sobre las líneas de Nasca durante dos horas seguidas, sobre estas aeronaves se instalaron los equipos que enviarían la información a la estación base situada en la torre de control del aeródromo de Nasca, esta ubicación se programo como Latitud y Longitud de referencia.

Para la estación central se transportaron equipos desde Lima, como dos receptores VHF AM marca Harris, cargadores de baterías, antenas omnidireccionales de alta ganancia, guías de onda Heliax de 1/2 pulgada de diámetro, dos computadoras de escritorio y una computadora portátil. Todo

este equipamiento fue instalado en la torre de control en forma paralela con los equipos transmisores en las aeronaves.

Posteriormente se programaron los vuelos con los controladores de tránsito aéreo del aeródromo de Nasca de tal forma que estos pudieran salir uno detrás de otro y poder observar en funcionamiento las tres aeronaves en el programa desarrollado en Visual Basic (programa desarrollado por Corpac), el cual recibe en forma serial la información de latitud y longitud de un determinado equipo.

Luego se energizaron los tres equipos que empezaron a capturar su información usando su correspondiente equipo receptor GPS y una vez que ubicaron su posición empezaron a enviar su posición a la estación central, dado que los receptores VHF AM poseen un parlante externo se podía escuchar las tramas que llegaban en forma de tonos audibles a la estación e inmediatamente esta información decodificada era enviada en forma serial a la computadora portátil que era donde se visualizaba.

Una vez que las aeronaves decolaron de la pista de aterrizaje, los equipos seguían enviando información de su posición actual que cambiaba constantemente, este hecho podía observarse en el programa de monitoreo como un movimiento de pequeños saltos.

Cada equipo estaba configurado con una identificación seleccionable por hardware mediante un selector (dip switch). Dependiendo de su identificación cada equipo podía identificar su turno de transmisión y es en ese momento que enviaba su información. Durante dos horas las tres

aeronaves realizaron el circuito de las líneas de Nasca y simulaban un aterrizaje para volver al circuito.

Finalmente una vez que se pudo verificar el funcionamiento y la factibilidad del proyecto una de las aeronaves se dirigió hacia el aeródromo de Ica de tal manera que se pudiese comprobar la cobertura. Efectivamente se pudo monitorear la posición de la aeronave hasta las 32 millas programadas, luego de lo cual si bien se tenía señal de radiofrecuencia el algoritmo no podía decodificar la posición. Este era un caso que se esperaba.

Finalmente en presencia de algunos funcionarios de CORPAC como el Jefe del Área de Proyectos Especiales de Aeronavegación, el Jefe de la Oficina de la OACI en el Perú, el Jefe del Área de Inspección en vuelo y los controladores de tránsito aéreo del aeródromo de Nasca se dio conformidad a las pruebas realizadas en dicho aeródromo.

Se decidió luego seguir perfeccionando el sistema de manera que un futuro se pueda implementar definitivamente en las aeronaves. El gerente de la compañía Aerocondor que opera en dicho aeródromo dio también su visto bueno al proyecto comprometiéndose a implementarlo cuando CORPAC así lo decida.

7.6 Análisis de mejoras futuras

7.6.1 Equipos GPS

Se hizo un análisis de las opciones que habían de integrar a nuestro diseño el equipo receptor GPS ya que hasta este punto este dispositivo sigue siendo un equipo externo.

Existen tarjetas receptoras que fácilmente se pueden integrar al prototipo en desarrollo y forme parte de un solo equipo en un mismo chasis y lo único externo sean las antenas y los subsistemas de energía eléctrica.

Tabla 7.7: Cuadro comparativo de tarjetas

Fabricante	MARCONI	MARCONI	AXIOMNAV	ASHTEC	IBM
Modelo	<i>All Star</i>	<i>Super Star</i>	<i>Swift B1</i>	<i>Sensor II Mag</i>	<i>IBM 43 GA</i>
Canales	12	12	12	12	12
Update Rate (seg)	1	1	1	1	1
Protocolo	NMEA	NMEA	NMEA	NMEA	NMEA/RTCM
Potencia-Corriente	1.4W-280Ma	1.2W-240mA	0.75W-150mA	1.7W-340mA	250mA
Tiempo readquisición	1 - 3 seg	1 - 3 seg	2 seg	2 seg	1 seg
Vel. Max. (nudos)	1 000	1 000	1 000	1 000	1 000
Altitud máxima (pies)	60 000	60 000	60 000	60 000	60 000
Cold Start (seg)	120	120	45	90	90
Warm Start (seg)	45	45	38	30	40
Hot Start (seg)	15	15	8		15
Puertos I/O	2 UART	2 UART	2 UART	2 UART	
Tamaño (mm)	67x102x14	46x71x13	26x32x13	106x57	66x40x4.5
Precisión WAAS (m)	3		10		
Precisión DGPS (m)	1	1	1 - 5		Si

S/A16 16 25
Off
(m)

S/A On (m)	40	40	100		
------------	----	----	-----	--	--

Descripción de los parámetros:

Canales: Se refiere a la cantidad de canales de recepción de satélites.

Update rate: Se refiere al tiempo mínimo en que se actualiza la información por el puerto serial.

Protocolo: Se refiere al protocolo usado para transmitir la información por el puerto serial.

Potencia-corriente: Indica el consumo de energía de la tarjeta receptora.

Tiempo de readquisición: Indica el tiempo que demora en readquirir un satélite en caso de que pierda su señal.

Velocidad máxima: Indica la máxima velocidad a la que puede ir el receptor y aun obtener una lectura real de su posición.

Altitud máxima: Se refiere a la altitud máxima en la que puede estar el receptor y aun obtener una lectura real de su posición.

Cold Start: Es el tiempo que se demora el receptor GPS en obtener información de su posición cuando se enciende este.

Warm Start: Es el tiempo que demora el receptor GPS en obtener su posición si es que recientemente obtuvo información de los satélites. Menos de 4 horas

Hot Start: Es el tiempo de actualización de la posición si es que por algunos segundos se perdió conexión con los satélites por ejemplo al pasar por un túnel.

Puertos I/O: Indica la cantidad de puertos seriales que posee la tarjeta.

Tamaño: Indica las dimensiones físicas de la tarjeta.

Precisión WAAS: Indica si posee la capacidad de precisión WAAS

Precisión DGPS: Indica si posee la capacidad de precisión DGPS

S/A Off: Indica que la opción de Select Availability esta deshabilitada.

S/A On: Indica que la opción de Select Availability esta habilitada.

Selective Availability. Mecanismo por medio del cual EE.UU. añade errores en las referencias enviadas por los satélites, con el fin de disminuir la precisión del sistema. Actualmente está desactivado.

7.6.2 Incremento de aeronaves y acceso al medio

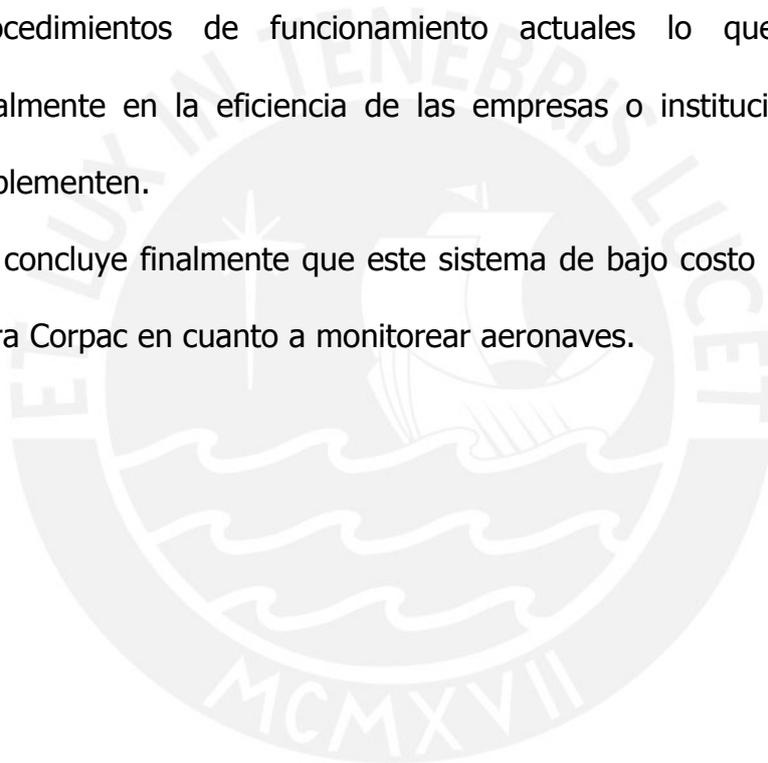
Como se mencionó durante el estudio del acceso al medio, para este proyecto en particular se confeccionaron 03 prototipos y el sistema de acceso al medio fue un sistema TDMA que sincroniza con la hora UTC del GPS. Para un futuro se requiere que la cantidad de aeronaves se incremente a 16 unidades, en este caso se tendrá que realizar un estudio y diseño de un sistema de acceso al medio que pueda soportar la cantidad de aeronaves precisadas.

El sistema de acceso podría cambiar a un sistema CSMA y se utilizaría la entrada que poseen los prototipos de detección de portadora y que en este caso no se utilizó.

8. CONCLUSIONES

1. Se concluye que es posible demostrar experimentalmente una idea surgida en un Laboratorio que fue la de transmitir tonos audibles a través de radios VHF con modulación AM y que estas pudieran ser decodificadas en el extremo receptor para aplicaciones en aviación civil, usando la banda aeronáutica.
2. Se concluye que los tonos duales multi-frecuencia pueden ser útiles no solamente como señalización entre centrales telefónicas, sino también como modulador de portadoras VHF AM para intercambio de información.
3. Se comprueba la utilidad del sistema GPS para aplicaciones civiles de monitoreo de posición geográfica.
4. Este sistema de transmisión de información es aplicable a muchos otros casos en el que se necesite transmitir datos usando medios relativamente baratos y accesibles sin necesidad de desarrollar complicados sistemas de modulación digital, aplicaciones en las que no se necesitan intercambiar mucho volumen de información, ni con tanta frecuencia como puede ser por ejemplo en telemetría para monitoreo de equipos tales como compresores, bombas, válvulas, etc.
5. Un sistema de este tipo se puede implementar para monitoreo por demanda de distintos tipos de estaciones móviles o fijas con las cuales se necesite intercambiar información.

6. Se pudo concretar en la segunda etapa el diseño y funcionamiento de un sistema de acceso al medio con multi-canalización en el dominio del tiempo. Se realizó para tres equipos pero que más adelante puede ser mejorado y ampliado a más estaciones.
7. Existen en el país muchos campos en los cuales podemos desarrollar proyectos, sistemas o prototipos de equipos que permitan mejorar los procedimientos de funcionamiento actuales lo que repercutirá finalmente en la eficiencia de las empresas o instituciones que los implementen.
8. Se concluye finalmente que este sistema de bajo costo es de utilidad para Corpac en cuanto a monitorear aeronaves.



9. OBSERVACIONES Y RECOMENDACIONES

Se recomienda proseguir con el desarrollo de este sistema, incrementando la cantidad de aeronaves o estaciones, ya que las ventajas de bajo costo de este sistema son apreciables.

Proyectos de este tipo pueden desarrollarse en nuestro país mediante el apoyo decidido de empresas comprometidas con la investigación y con presupuestos que permitan el desarrollo sin contratiempos de lo planificado.



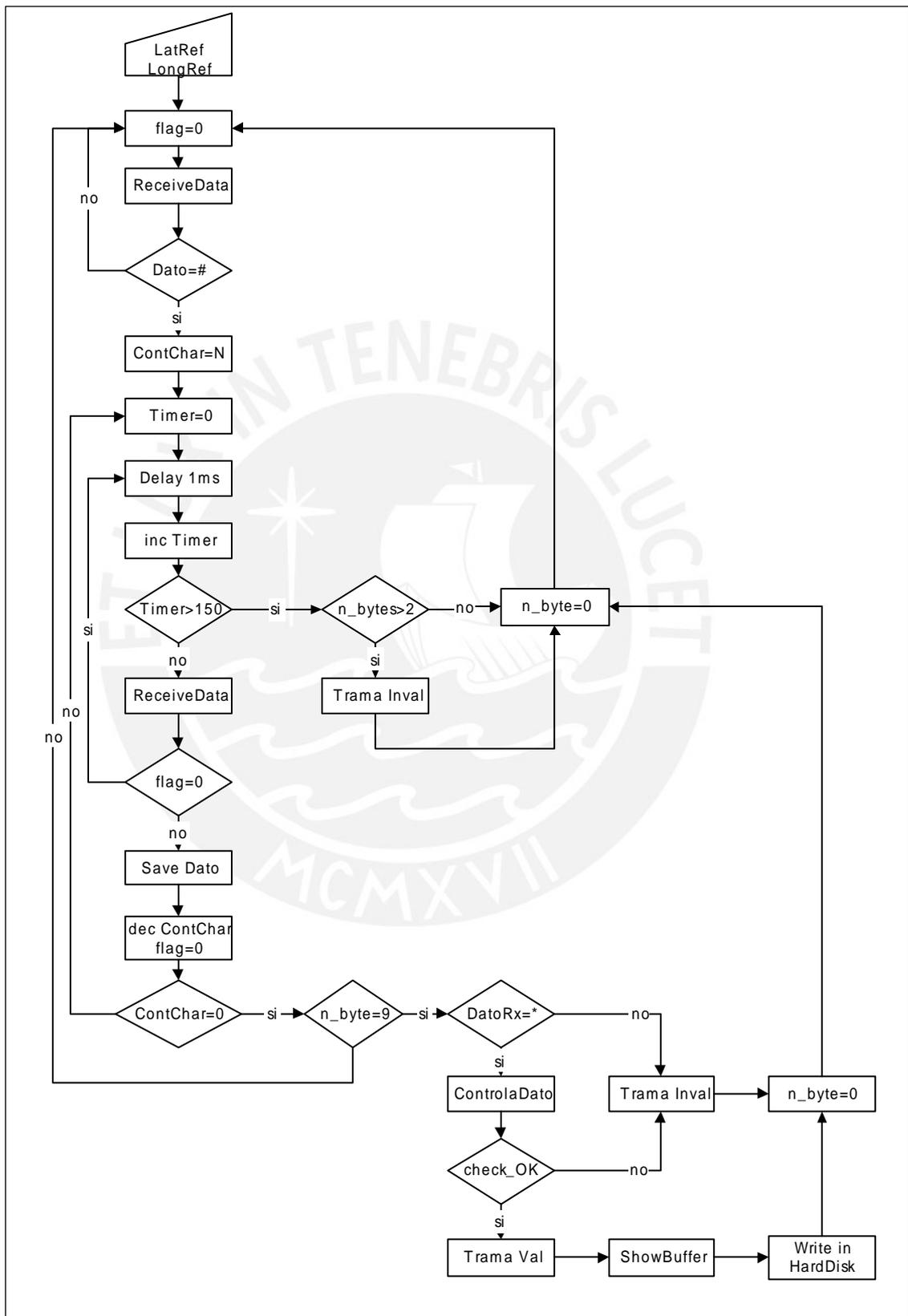
BIBLIOGRAFIA

1. CORPAC
1999 GNSS Sistema Mundial de Navegación por Satélite
2. TOMASI, Wayne
1996 Sistemas de Comunicaciones Electrónicas Segunda Edición
3. MICROCHIP
Manual de Microcontrolador PIC16F877
4. MITEL
Hoja de datos del 5089
5. MITEL
Hoja de datos del 8870
6. MITEL
Hoja de datos del 8888
7. Francisco Ramos Pascual
http://www.radioptica.com/Radio/intermodulacion_I.asp?pag=2
8. <http://www.pangolinsms.com/tech03-types-of-mobile-phones.htm>
9. <http://www.networkdictionary.com/telecom/fdm.php>

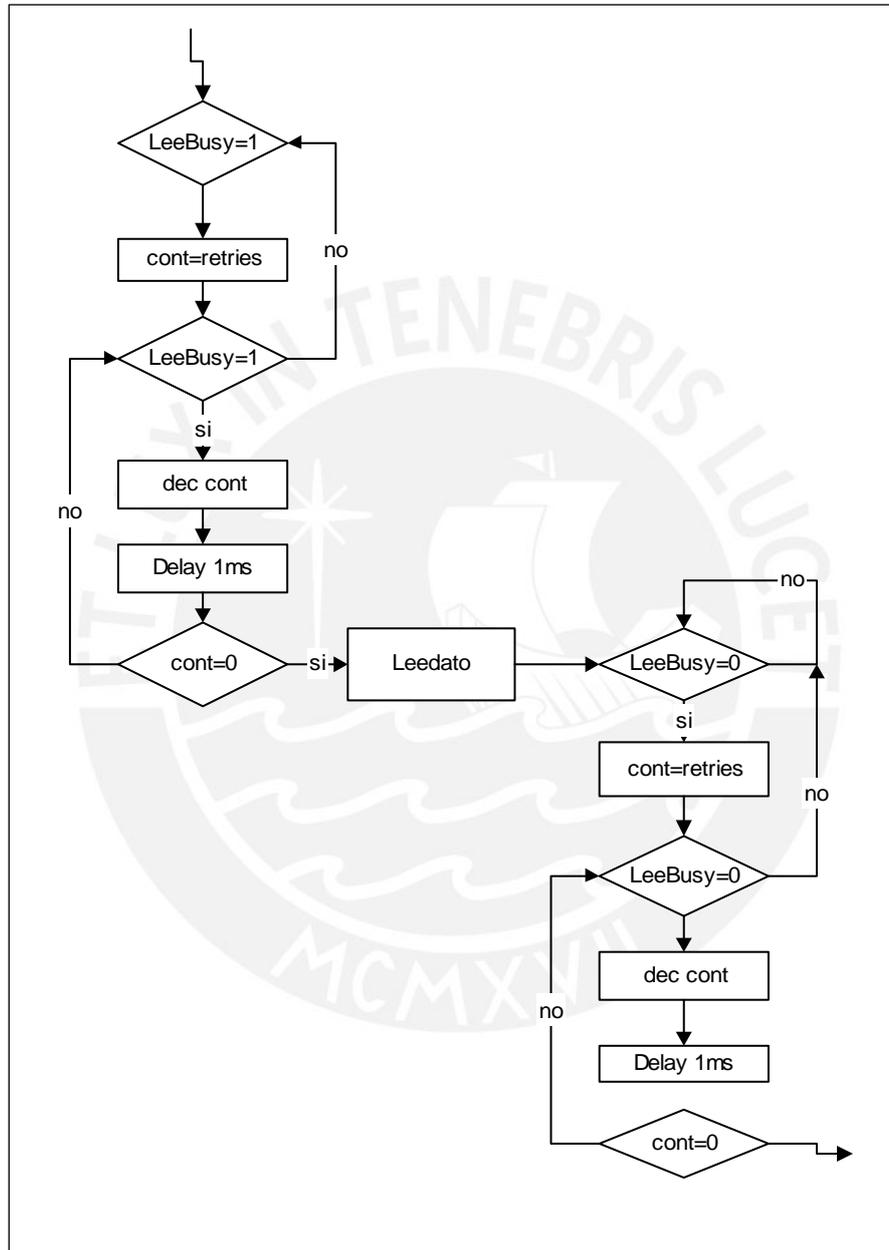


ANEXO 1: DIAGRAMAS DE FLUJO DE LOS PROGRAMAS

2.2 DIAGRAMA DE FLUJO DEL PROGRAMA DE RECEPCIÓN



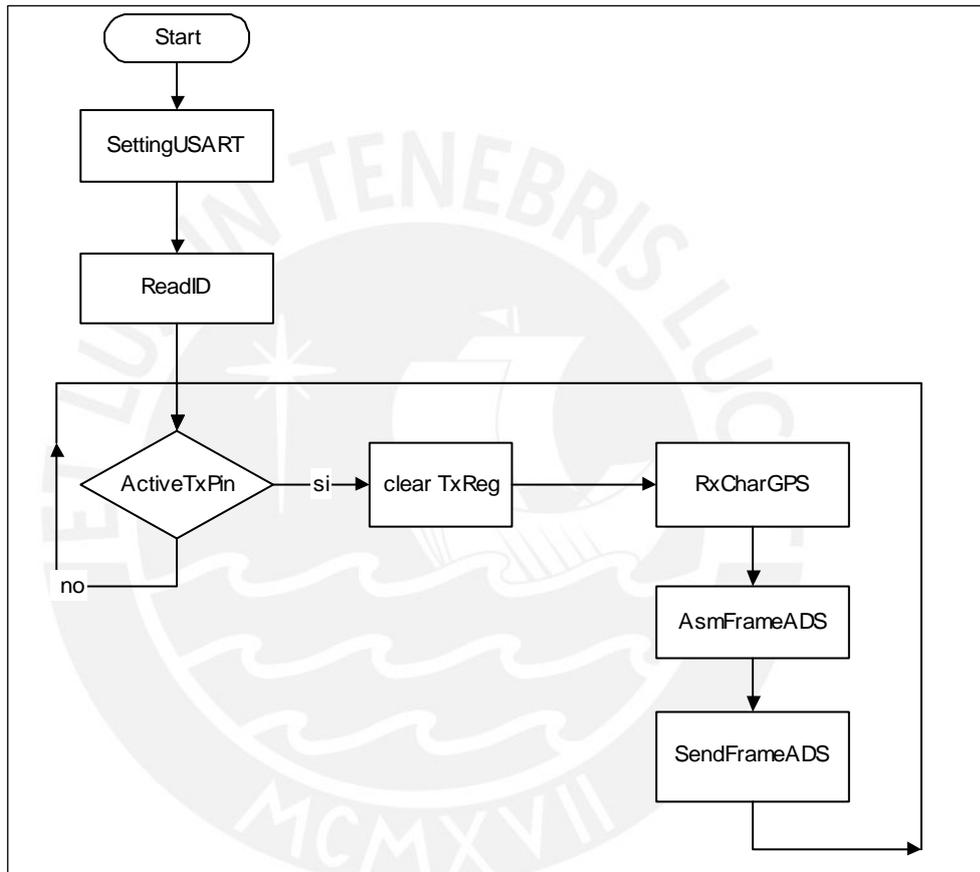
b. ReceiveData



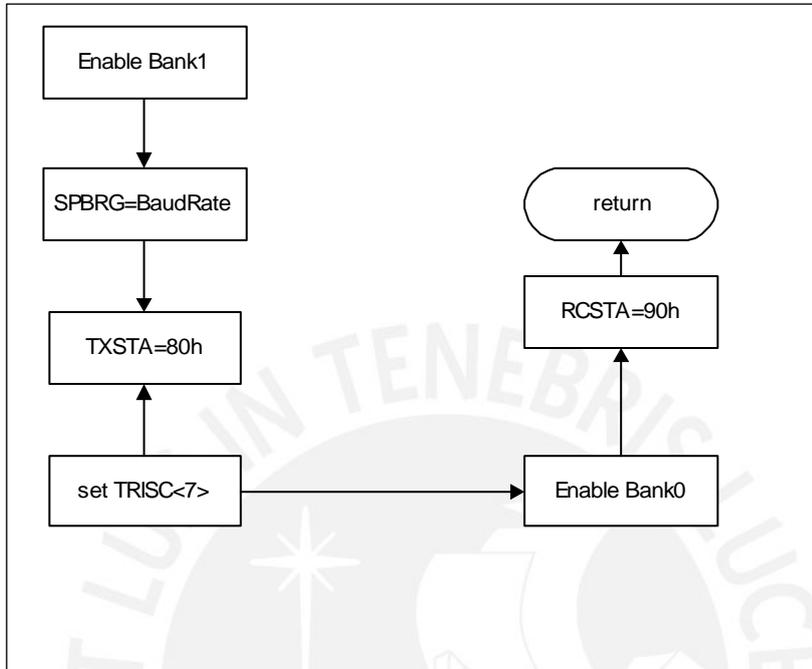
- LeeBusy es una rutina que sondea el pin que indica si que se ha detectado un tono DTMF.
- LeeDato es una rutina que lee el dato del puerto paralelo.

3.2 DIAGRAMA DE FLUJO DEL PROGRAMA DE TRANSMISION

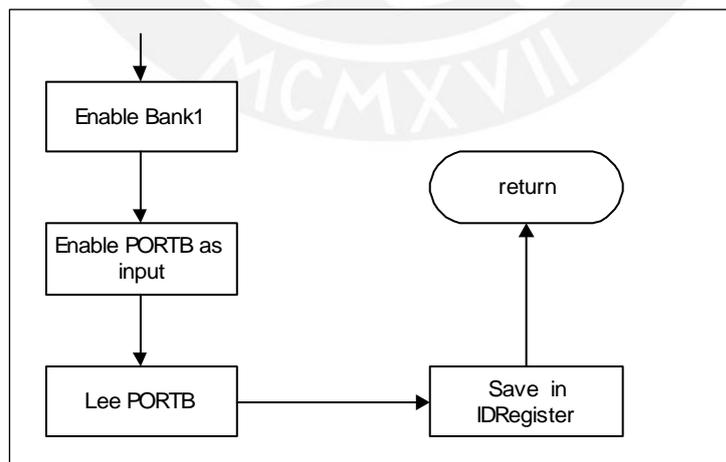
1. Inicio



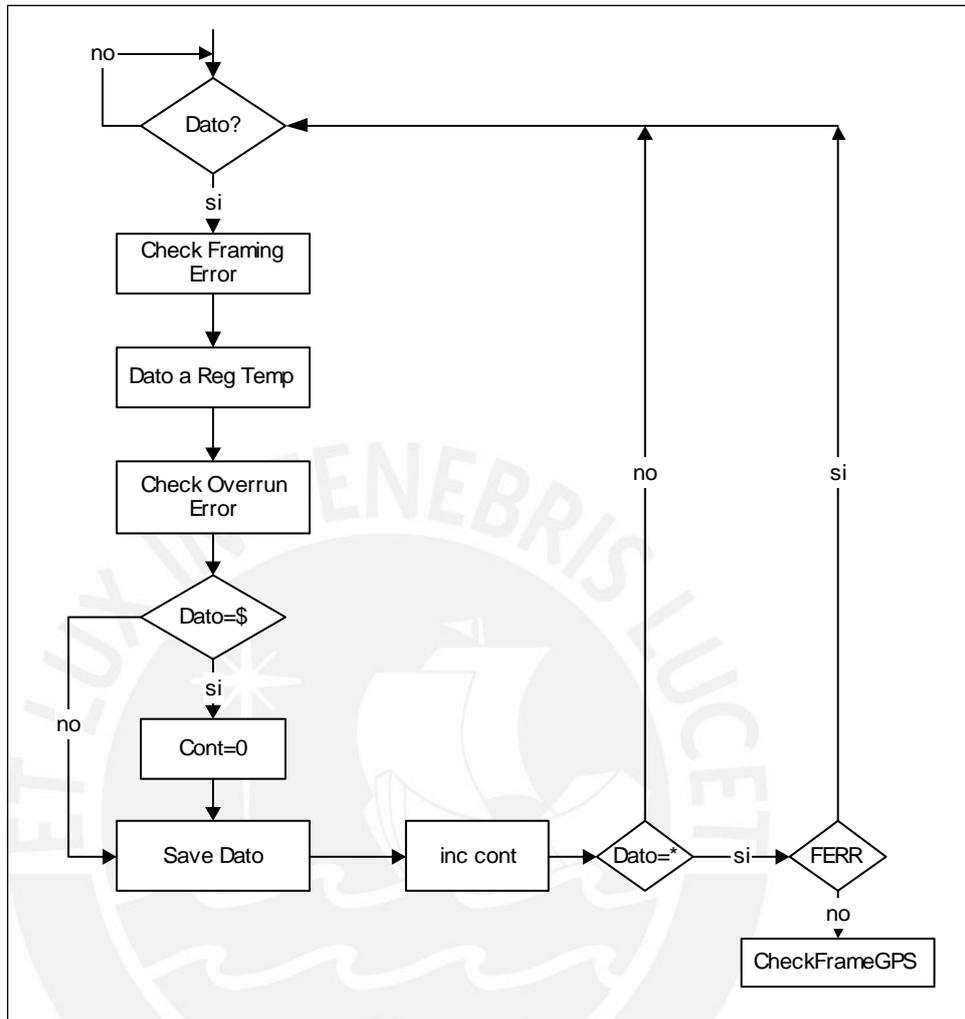
1.1 Inicialización del USART (*SettingUSART*)



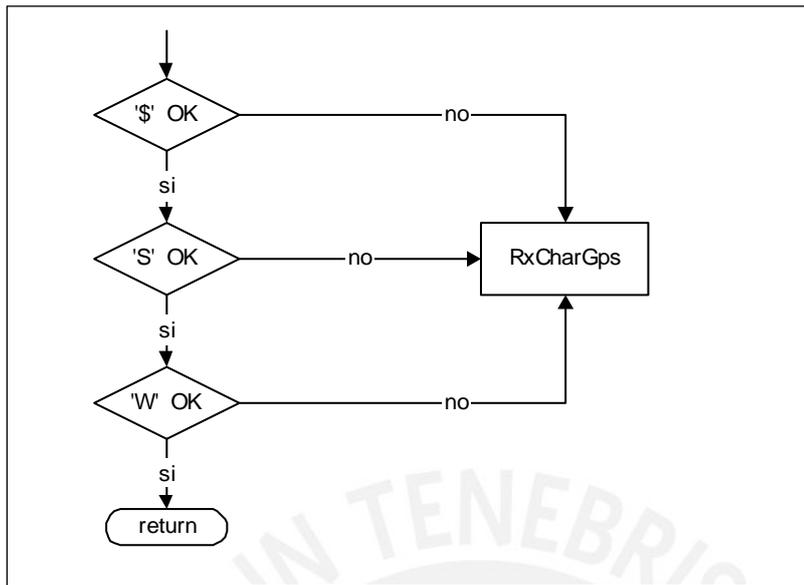
1.2 Lectura de Identificación (*ReadID*)



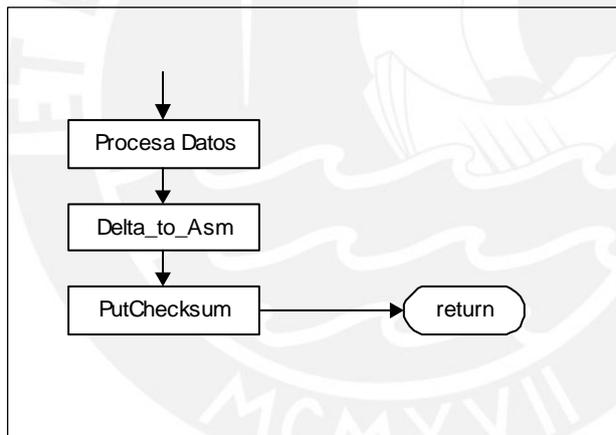
1.3 Recepción de caracteres del GPS (*RxCharGPS*)



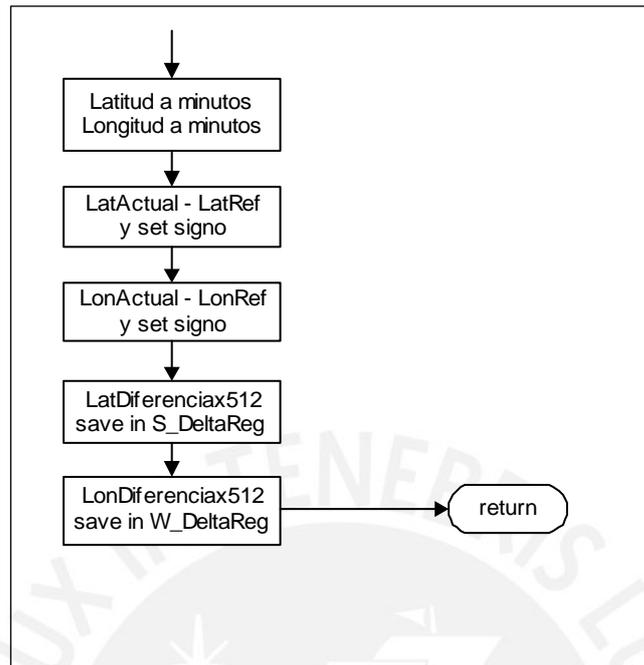
1.3.1 Chequeo de la trama GPS recibida (*CheckFrameGPS*)



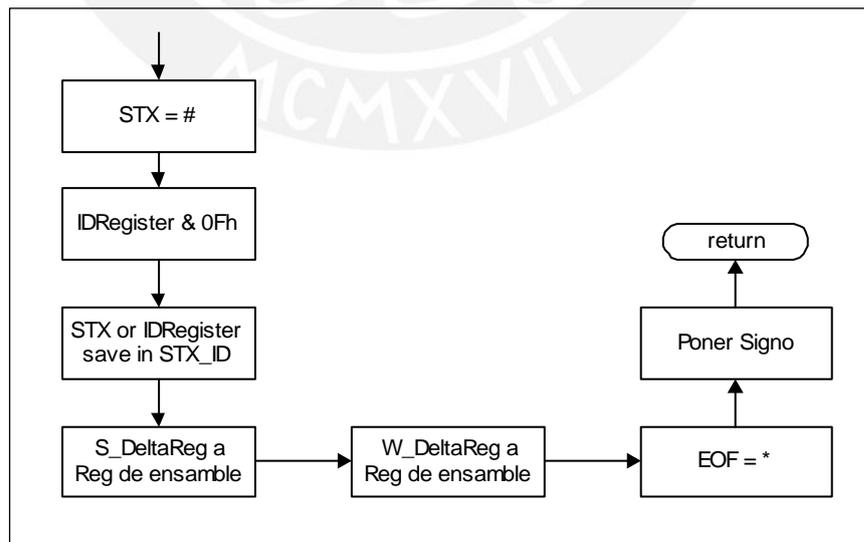
1.4 Proceso y ensamble de la trama ADS (*AsmFrameADS*)



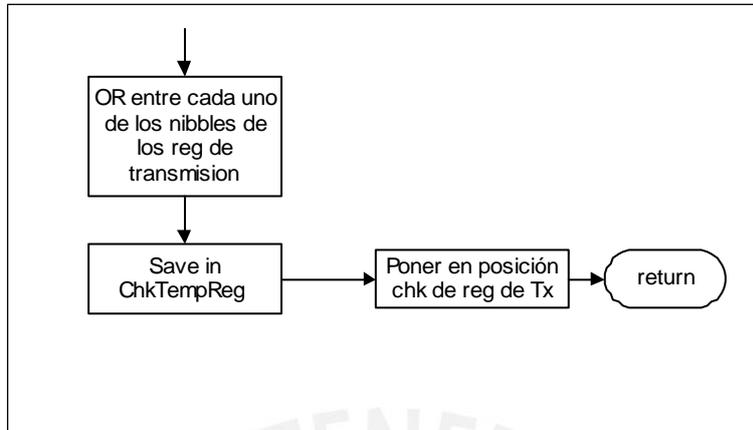
1.4.1 Cálculo del Delta de posición (*ProcesaDatos*)



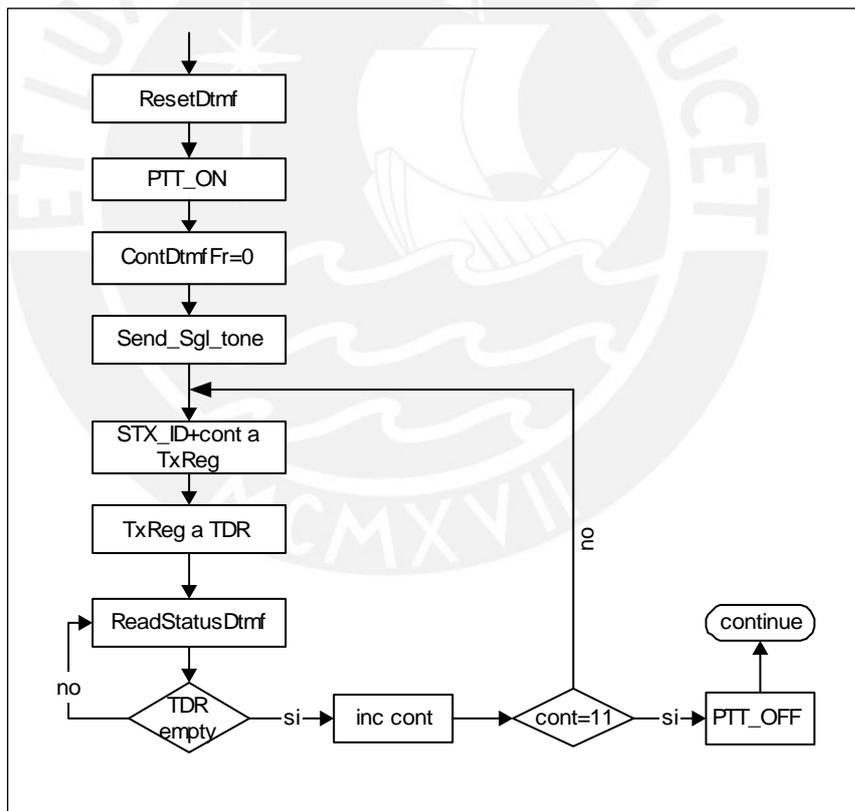
1.4.2 Pone delta en registros de transmisión (*Delta_to_Asm*)



1.4.3 Calcula y pone el checksum (*PutChecksum*)

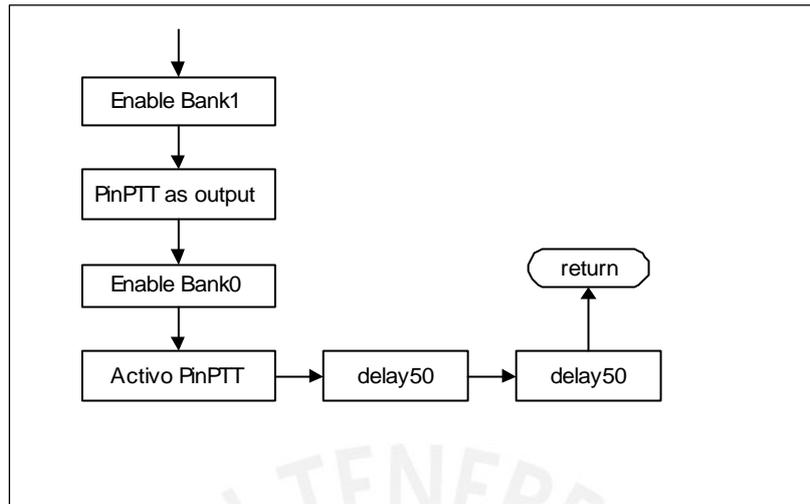


1.5 Envío de la trama ADS mediante tonos DTMF (*SendFrameADS*)

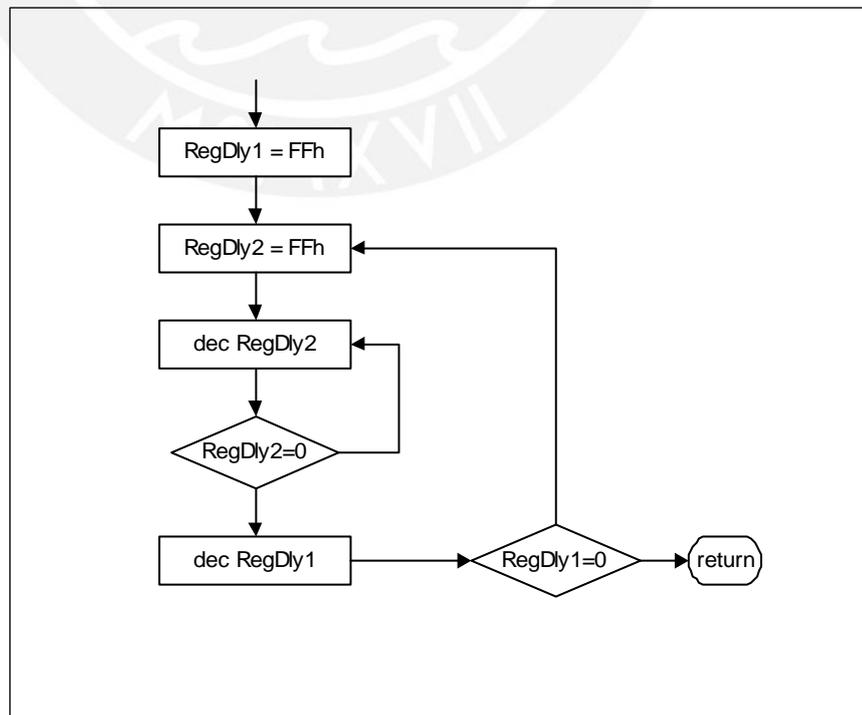


1.5.1 ResetDtmf :Rutina de inicialización del IC Dtmf M8888

1.5.2 Activa el PTT y retarda 100 ms (*PTT_ON*)



1.5.2.1 Rutina que retarda 50ms (*delay50*)





Anexo B. Programa en Assembler

```

;*****
;PROGRAMA DE TRANSMISION DTMF
;Creado por Ricardo Cárdenas Abanto para CORPAC S.A. - APEA
;Version 1.0
;*****
        list p=16F877
        include "Ricardo.inc"
;*****Reg bank 0 *****
RegBaseA equ 20h
        cblock   RegBaseA

        endc
;=====
cblock   0x73
        RegDly1
        RegDly2
        IDRegister
        mulcnd       ; 8 bit multiplicando
        mulplr       ; 8 bit multiplicador
        Prod_H       ; High byte of the 16 bit result
        Prod_L       ; Low byte of the 16 bit result
        count_mul
        Dtmf_St_Reg  ;Reg de status del 8888
        RDR          ;receive data register 8888
        StrTempReg   ; para la rutina de gps
        TxReg        ;registro temporal de tx
        ChkTempReg

        endc
;=====
        cblock   0x20
TEMP7
TEMP8
MATHREG1:4           ;math routine registers - 32 bits
MATHREG2:4           ;32 bits
MATHREG3:8           ;64 bits - usado en rutina de mult
MATHFLAG             ;flag register
MATHSIGN             ;sign register
MATHREG4:4           ;registro temporal
S_Delta_Reg:4        ;reg temporal de ensamble
W_Delta_Reg:4        ; idem
LatRefReg:2          ;reg de referencia
LonRefReg:2          ; idem
STX                  ;reg temporal Start Tx
STX_ID               ;REGISTROS DE ENSAMBLE
LAT13_6              ;DE LA ADS FRAME
LAT5_0_LON13_12     ;
LON11_4              ;
LON3_0_CHK           ;
EOF_NULL             ;
SIGNLAT              ;aqui set el signo de latitud
SIGNLON              ;aqui set el signo de longitud
        endc
;*****constantes ref*****
latref equ 0x0375     ;1445 en minutos
lonref equ 0x1196     ;7502 en minutos
;*****Constantes para DTMF*****
pinPTT equ 0x00       ;pin 0 del portc
variable count_DtmfFr
variable count_temp

```

```

sel_Status      equ  B'00000101'  ;status reg del 8888
sel_Control     equ  B'00000110'  ;controlreg del 8888
sel_TDR         equ  B'00000010'
sel_RDR         equ  B'00000001'
;*****Constantes para el USART*****
TXSTA_bits      equ  0x80          ;CSRC=1
RCSTA_bits      equ  0x90          ;SPEN=1 , CREN=1
BaudRate        equ  0x1F          ;9600bps osc 20Mhz;0x19 con
16Mhz
variable        ByteCount=0
ActiveTxPin     equ  0x04          ;push Send
;*****Trama GPS*****
PosDollarId     equ  .0
PosSouthId      equ  .23
PosWestId       equ  .35
variable        count_GPSFr
;*****
#define          BANK1 bsf STATUS,RP0 ;Select BANK1
#define          BANK0 bcf STATUS,RP0 ;Select BANK0
Org 0
goto Start
;*****MOVER*****
MOVER macro FROM,TO
movf FROM,W
movwf TO
endm
;***** Rutina de MUL *****
;*****N1 en mulcnd y N2 en mulplr *****
;*****resultados en Prod_H y en Prod_L*****
mpy:   clrf   Prod_H
       clrf   Prod_L
       movlw 8
       movwf count_mul
       movf  mulcnd, W
       bcf  STATUS, C          ; Clear carry bit
loop   rrf   mulplr, F
       btfsc STATUS, C
       addwf Prod_H, F
       rrf   Prod_H, F
       rrf   Prod_L, F
       decfsz count_mul, F
       goto loop
       retlw 0
;*****Configuracion del USART*****
SettingUSART:
       bcf   STATUS,RP1
       BANK1
       movlw BaudRate
       movwf SPBRG
       movlw TXSTA_bits
       movwf TXSTA
       bsf   TRISC,7
       BANK0
       movlw RCSTA_bits
       movwf RCSTA
       return

;*****Lee identificacion y activacion de transmision*****
;*****del PORTB*****
;*****y lo guarda en IDRegister*****
ReadID:  BANK1

```

```

movlw 0x1F          ; pines 0,1,2,3,4 as input
movwf TRISB
BANK0
clrf IDRegister
movf PORTB,W
movwf IDRegister
return
;*****Check y reset del Overrun error*****
Check_OERR: btfss RCSTA,OERR          ;check OERR
            goto Out_OERR
            bcf RCSTA,CREN
            bsf RCSTA,CREN
Out_OERR:   return
;*****Check FERR*****
Check_FERR:
            btfsc RCSTA,FERR          ;check FERR
            bsf TxReg,0
            return
;*****Rutina de Rx de Byte del GPS*****
RxCharGPS: btfsc PIR1,RCIF          ;test RCIF
            goto ReadChar
            goto RxCharGPS
ReadChar:
            call Check_FERR          ;check si FERR set
            MOVER RCREG,StrTempReg
            call Check_OERR          ;check y clear si OERR
set
            movf StrTempReg,W
            sublw 0x24
            btfss STATUS,Z          ;if RxByte=='$' ; 24
ByteCount=0
            goto StoreChar
StoreChar: MOVER StrTempReg,RegBaseA+ByteCount
ByteCount++
            movlw 0x2A
            subwf StrTempReg,W
            btfsc STATUS,Z          ;if RxByte=='*' ; 2A
            goto RxCharGPS
            goto CheckErrorRxGPS
CheckErrorRxGPS: btfsc TxReg,0
                goto RxCharGPS
                goto CheckFrameGPS
;*****Chequea la trama Recibida del GPS*****
CheckFrameGPS: movf RegBaseA+PosDollarId,W
                sublw 0x24          ;if RegTempComp=='$'
                btfsc STATUS,Z
                goto SecondComp
                goto RxCharGPS
SecondComp   movf RegBaseA+PosSouthId,W
                sublw 0x53
                btfsc STATUS,Z          ;if
RegTempComp=='S'
                goto ThirdComp
                goto RxCharGPS
ThirdComp   movf RegBaseA+PosWestId,W
                sublw 0x57
                btfsc STATUS,Z          ;if
RegTempComp=='W'
                goto IfGPSFr_OK

```

```

                                goto RxCharGPS                                ;else goto
RxCharGPS                                                                ;endif
IfGPSFr_OK:    return
;*****
Init_Regs_Proceso macro
    clrf  SIRH
    clrf  SIRL
    movlw 0x64
    movwf SWRC
    movlw 0x0A
    movwf SWRD
    movlw 0x01
    movwf SWRU
    clrf  SDRH
    clrf  SDRL
    clrf  WDRH
    clrf  WDRL
endm
Init_Regs_Ref macro
    movlw high latref
    movwf LatRefReg
    movlw low latref
    movwf LatRefReg+1
    movlw high lonref
    movwf LonRefReg
    movlw low lonref
    movwf LonRefReg+1
    clrf  SIGNALAT
    clrf  SIGNALON
endm
;*****
PutToFile macro  xregL,xregH
    movf  Prod_L,W
    addwf xregL,F
    btfsc STATUS,C
    incf  xregH,f
    movf  Prod_H,W
    addwf xregH,F
endm
;*****
;subc macro que resta dos bytes con borrow
;input:    byte1
;          byte2
;          MATHFLAG,0 = borrow in
;output:byte1          = byte1 - byte2 - borrow in
;          MATHFLAG,0 = borrow out
subc macro byte2, byte1
    movf  MATHFLAG,W
    clrf  MATHFLAG
    subwf byte1,F
    btfss STATUS,C
    bsf  MATHFLAG,0
    movf  byte2,W
    subwf byte1,F
    btfss STATUS,C
    bsf  MATHFLAG,0
endm
;*****multiplica dos registros*****
MULF macro  RegM,Regm
    movf  RegM,W
    movwf mulcnd

```

```

        movf    Regm,W
        movwf   mulplr
        goto   mpy
    endm
;*****
MULC    macro   Cte,Regm
        movlw   Cte
        movwf   mulcnd
        movf    Regm,W
        movwf   mulplr
        goto   mpy
    endm
;*****C2*****
SCOMP2: comf   SIRL,F
        comf   SIRH,F
        incf   SIRL,F
        btfsc  STATUS,Z
        incf   SIRH,F
        bsf    SIGNLAT,0
        return
;
WCOMP2: comf   WIRL,F
        comf   WIRH,F
        incf   WIRL,F
        btfsc  STATUS,Z
        incf   WIRH,F
        bsf    SIGNLON,0
        return
;*****
Llena_mulcdr macro
        ; 512
        movlw  0x00
        movwf  MATHREG2
        movlw  0x00
        movwf  MATHREG2+1
        movlw  0x02
        movwf  MATHREG2+2
        movlw  0x00
        movwf  MATHREG2+3
    endm
;*****
Llena_Mulcnd macro REGMH,REGML
        movlw  0x00
        movwf  MATHREG1
        movlw  0x00
        movwf  MATHREG1+1
        movf   REGMH,W
        movwf  MATHREG1+2
        movf   REGML,W
        movwf  MATHREG1+2
    endm
;*****
Divisor_eq_mil macro
        movlw  0x00
        movwf  MATHREG2
        movlw  0x00
        movwf  MATHREG2+1
        movlw  0x03
        movwf  MATHREG2+2
        movlw  0xE8
        movwf  MATHREG2+3

```

```

endm
;*****
movltoX macro    MATHREGX
    movf        MATHREG1+3,W
    movwf       MATHREGX+3
    movf        MATHREG1+2,W
    movwf       MATHREGX+2
    movf        MATHREG1+1,W
    movwf       MATHREGX+1
    movf        MATHREG1,W
    movwf       MATHREGX
endm

;*****Ensambla la trama ADS*****
;*****
AsmFrameADS call ProcesaDataos
              call Delta_to_Asm
              call PutChecksum
              return

;*****
ProcesaDataos:
    Init_Regs_Proceso
;    -----para latitud-----
count_GPSFr=.14
    MULC    0x0A,RegBaseA+count_GPSFr
    PutToFile SIRL,SIRH
    movf    RegBaseA+count_GPSFr+1,W
    addwf   SIRL,F
    MULC    0x3C,SIRL
    clrf    SIRH
    clrf    SIRL
    PutToFile SIRL,SIRH
    MULC    0x0A,RegBaseA+count_GPSFr+2
    PutToFile SIRL,SIRH
    MULC    0x01,RegBaseA+count_GPSFr+3
    PutToFile SIRL,SIRH
; hasta aqui tengo latactual en minutos
;siguen los decimales
count_GPSFr=.19
    MULF    SWRC,RegBaseA+count_GPSFr
    PutToFile SDRL,SDRH
count_GPSFr++
    MULF    SWRD,RegBaseA+count_GPSFr
    PutToFile SDRL,SDRH
count_GPSFr++
    MULF    SWRU,RegBaseA+count_GPSFr
    PutToFile SDRL,SDRH
    MOVER   SDRL,SDRRL
    MOVER   SDRH,SDRRH
;    -----para longitud-----
count_GPSFr=.26
    MULC    0x0A,RegBaseA+count_GPSFr
    PutToFile WIRL,WIRH
    movf    RegBaseA+count_GPSFr+1,W
    addwf   WIRL,F
    MULC    0x3C,WIRL
    clrf    WIRH
    clrf    WIRL
    PutToFile WIRL,WIRH
    MULC    0x0A,RegBaseA+count_GPSFr+2

```

```

        PutToFile WIRL,WIRH
        MULC    0x01,RegBaseA+count_GPSFr+3
        PutToFile WIRL,WIRH
; hasta aqui tengo lonactual en minutos
;ahora los decimales
count_GPSFr=.31
        MULF   SWRC,RegBaseA+count_GPSFr
        PutToFile  WDRL,WDRH
count_GPSFr++
        MULF   SWRD,RegBaseA+count_GPSFr
        PutToFile  WDRL,WDRH
count_GPSFr++
        MULF   SWRU,RegBaseA+count_GPSFr
        PutToFile  WDRL,WDRH
        MOVER  WDRL,WDRRL
        MOVER  WDRH,WDRRH
;-----
;la diferencia sera latactual-ref para evitar restar decs
        Init_Regs_Ref
        clrf   MATHFLAG
        subc   SIRL,LatRefReg+1
        subc   SIRH,LatRefReg
        btpsc  SIRH,7

        call   SCOMP2
        MOVER  SIRL,SIRRL
        MOVER  SIRH,SIRRH

;la diferencia sera lonactual-ref para evitar restar decs
        clrf   MATHFLAG
        subc   WIRL,LonRefReg+1
        subc   WIRH,LonRefReg
        btpsc  WIRH,7
        call   WCOMP2
        MOVER  WIRL,WIRRL
        MOVER  WIRH,WIRRH
        goto   Dif_to_delta

;*****
Dif_to_delta  Llana_mulcdr
              Llana_Mulcnd SIRRH,SIRRL
              call  mult12to3
              call  mov3to4           ; M4=entero*512
              Llana_mulcdr
              Llana_Mulcnd SDRRH,SDRRL
              call  mult12to3
              Divisor_eq_mil
              call  div32to1         ; hasta aqui en
M1=(decimal*512)/1000
              call  mov4to2           ;mover 4 a 2
              call  add2to1           ; en M1 el num solo falta round
              movltoX S_Delta_Reg
              ;FALTA EL REDONDEO
              ;-----
              Llana_mulcdr
              Llana_Mulcnd WIRRH,WIRRL
              call  mult12to3
              call  mov3to4           ; M4=entero*512
              Llana_mulcdr
              Llana_Mulcnd WDRRH,WDRRL
              call  mult12to3

```

```

    Divisor_eq_mil
    call div32to1          ; aqui M1=(decimal*512)/1000
    call mov4to2          ; mover 4 a 2
    call add2to1          ; en M1 el num falta round
    movltoX W_Delta_Reg
    ;FALTA EL REDONDEO
    return

Delta_to_Asm: BANK0
    movlw 0xC0            ; 12(0Ch) = # en DTMF
    movwf STX
    movf IDRegister,W
    andlw 0x0F
    iorwf STX,W
    movwf STX_ID ; hasta aqui la cabecera y el ID
    rlf W_Delta_Reg+3,F
    rlf W_Delta_Reg+2,F
    rlf W_Delta_Reg+3,F
    rlf W_Delta_Reg+2,F
    rlf W_Delta_Reg+3,F
    rlf W_Delta_Reg+2,F
    rlf S_Delta_Reg+3,F
    rlf S_Delta_Reg+2,F
    rlf W_Delta_Reg+3,F
    rlf W_Delta_Reg+2,F
    rlf S_Delta_Reg+3,F
    rlf S_Delta_Reg+2,F
    MOVER S_Delta_Reg+2,LAT13_6
    MOVER S_Delta_Reg+3,LAT5_0_LON13_12
    MOVER W_Delta_Reg+2,LON11_4
    MOVER W_Delta_Reg+3,LON3_0_CHK
    movlw 0xB0
    movwf EOF_NULL      ; FIN DE TRAMA
    call PutSign

    return
;*****
PutSign:    btfsc SIGNLAT,0
            bsf LAT13_6,7
            btfsc SIGNLON,0
            bsf LAT5_0_LON13_12,1
            return
;*****
PutChecksum: movf STX_ID,W
             xorwf STX_ID+1,W
             xorwf STX_ID+2,W
             xorwf STX_ID+3,W
             movwf ChkTempReg
             swapf ChkTempReg,W
             xorwf STX_ID+1,W
             xorwf STX_ID+2,W
             xorwf STX_ID+3,W
             xorwf STX_ID+4,W
             movwf ChkTempReg
             swapf ChkTempReg,F
             movlw 0xF0
             andwf STX_ID+4,F
             movlw 0x0F
             andwf ChkTempReg,W
             iorwf STX_ID+4,F
             return
;***** Rutina de retardo de 50 ms*****

```

```

delay50 movlw 0xFF
        movwf RegDly1
loop1   movlw 0xFF
        movwf RegDly2
loop2   decfsz RegDly2,F
        goto loop2
        decfsz RegDly1,F
        goto loop1
        return
;*****Enable PORTD for Write*****
WRPortD_ON macro
        BANK1
        movlw B'0000000' ;portd output pines(7,6,5,4 datos)
        movwf TRISD
        BANK0
endm

;*****Enable PORTD for Read*****
RDPortD_ON macro
        BANK1
        movlw B'11110000' ;portd input pines(7,6,5,4)
        movwf TRISD
        BANK0
endm

;*****hace y deshace PTT*****
PTT_ON   macro
        BANK1
        bsf TRISC,7
        BANK0
        bsf PORTC,pinPTT
        goto delay50
        goto delay50
endm
;
PTT_OFF  macro
        BANK0
        bcf PORTC,pinPTT
endm

;*****Software de reset del 8888*****
RdStatus macro
        WRPortD_ON
        movlw sel_Status ;
        movwf PORTD
        RDPortD_ON
        movf PORTD,W
        movwf Dtmf_St_Reg
endm
WrControl macro Bit_sel
        WRPortD_ON
        movlw sel_Control ;
        movwf PORTD
        movlw Bit_sel
        andlw 0xF0
        iorwf sel_Control,W
        movwf PORTD ; REVISAR!!!!!!!!!!!!
endm
ResetDtmf macro
        RdStatus
        WrControl 0x00
        WrControl 0x00
        WrControl 0x80

```

```

        WrControl 0x00
        RdStatus
    endm
;*****
WrL_TDR    macro bit_sel
    WRPortD_ON
    movlw sel_TDR
    movwf PORTD
    movlw bit_sel
    andlw 0xF0
    iorwf sel_TDR,W
    movwf PORTD
endm
WrF_TDR    macro filetx
    WRPortD_ON
    movlw sel_TDR
    movwf PORTD
    movf filetx,W
    andlw 0xF0
    iorwf sel_TDR,W
    movwf PORTD
endm

Rd_RDR    macro
    WRPortD_ON
    movlw sel_RDR
    movwf PORTD
    RDPortD_ON
    movf PORTD,W
    movwf RDR
endm
;*****Rutina que envia un single tone*****
Send_Sgl_tone macro
    WrControl B'11010000'    ;b7 b6 b5 b4
    WrControl B'11010000'
    WrL_TDR    0x30
    goto delay50
    goto delay50
    goto delay50
    goto delay50

endm
;*****
SendFrameADS:
    ResetDtmf                ;PWR_UP_8888
    PTT_ON
count_DtmfFr=0
    Send_Sgl_tone                ;send portd FreqSingle
    ;protocolo de 8888
SendChrDtmf    MOVER STX_ID+count_DtmfFr,TxReg
    WrF_TDR TxReg
Wait1TDR      RdStatus
    btfss Dtmf_St_Reg,1        ; TDR empty
    goto Wait1TDR
count_DtmfFr++
count_temp=0x0B-count_DtmfFr
    btfsc STATUS,Z
    goto EndSendFrameADS
    MOVER STX_ID+count_DtmfFr-1,TxReg
    swapf TxReg,F

```

```

        WrF_TDR TxReg
Wait2TDR   RdStatus
           btfss Dtmf_St_Reg,1 ; TDR empty
           goto Wait2TDR
           goto SendChrDtmf
           PTT_OFF
EndSendFrameADS: goto continue
;*****
Start      call  SettingUSART
           call  ReadID
continue   btfss IDRegister,ActiveTxPin
           goto  continue
           clrf  TxReg
           call  RxCharGPS
           call  AsmFrameADS
           call  SendFrameADS

;=====
;Rutinas matematicas
;=====
;addc macro que suma dos bytes con carry
;input:      byte1
;           byte2
;           MATHFLAG,0 = carry in
;output:byte1      = byte1 + byte2 + carry in
;           MATHFLAG,0 = carry out
addc macro byte2, byte1
        movf  byte2,W
        addwf MATHFLAG,W
        clrf  MATHFLAG
        btfsc STATUS,C
        bsf   MATHFLAG,0
        addwf byte1,F
        btfsc STATUS,C
        bsf   MATHFLAG,0
        endm
;
;
;
;
;add2to1: MATHREG1 = MATHREG1 + MATHREG2
;input:      MATHREG1 (32 bit signed binary)
;           MATHREG2 (32 bit signed binary)
;output:MATHREG1 (32 bit signed binary)
;           MATHFLAG = 0 if no overflow, 1 if overflow
add2to1:clrf      MATHFLAG
        addc  MATHREG2+3, MATHREG1+3
        addc  MATHREG2+2, MATHREG1+2
        addc  MATHREG2+1, MATHREG1+1
        addc  MATHREG2, MATHREG1
        return
;
sub2fr1:bcf MATHFLAG,0
        subc  MATHREG2+3, MATHREG1+3
        subc  MATHREG2+2, MATHREG1+2
        subc  MATHREG2+1, MATHREG1+1
        subc  MATHREG2, MATHREG1
        return
;
;

```

```

;mult12to3: MATHREG3 = MATHREG1 * MATHREG2.
;input:     MATHREG1 (32 bit signed binary)
;          MATHREG2 (32 bit signed binary)
;output:MATHREG3 (64 bit signed binary)
;regs:     MATHSIGN
;          TEMP7
;          TEMP8
mult12to3:
  clrf MATHSIGN      ;check si producto es +/-
  movf MATHREG1,W
  xorwf MATHREG2,W
  andlw 80
  btfss STATUS,Z    ;prod + if ambos igual signo
  bsf  MATHSIGN,0   ; else producto es -
  ;
  btfsc MATHREG1,7  ;hacer mathreg1 positivo
  call comp1
  btfsc MATHREG2,7  ;hacer mathreg2 positivo
  call comp2
  ;
  call clear3       ;mathreg3 = 0
  ;
  movlw .32         ;32 bits
  movwf TEMP8
mult_10:btfss      MATHREG2,7
  goto mult_20
  clrf MATHFLAG     ;mathreg3 = mathreg3 + mathreg1
  addc MATHREG1+3, MATHREG3+7
  addc MATHREG1+2, MATHREG3+6
  addc MATHREG1+1, MATHREG3+5
  addc MATHREG1, MATHREG3+4
  clrf TEMP7       ;continue add bit carry
  addc TEMP7, MATHREG3+3
  addc TEMP7, MATHREG3+2
  addc TEMP7, MATHREG3+2
  addc TEMP7, MATHREG3
mult_20:decf      TEMP8,F
  btfsc STATUS,Z
  goto mult_30     ;si 32 veces, goto mult_30
  call shift_left3
  call shift_left2
  goto mult_10
  ;
mult_30:btfsc     MATHSIGN,0
  call comp3
  return
;
;div32to1: MATHREG1 = MATHREG3 / MATHREG2.
;input:     MATHREG3 (64 bit signed binary)
;          MATHREG2 (32 bit signed binary)
;output:MATHREG1 (32 bit signed binary)
;          MATHREG3 most 4 bytes = remainder
;regs:     MATHSIGN
;          TEMP7
;          TEMP8
div32to1:
  clrf MATHSIGN     ;check si prod +/-
  movf MATHREG2,W
  xorwf MATHREG3,W
  andlw 80
  btfss STATUS,Z   ;prod + if ambos igual signo

```

```

    bsf    MATHSIGN,0 ; else producto es -
    ;
    btfsc  MATHREG2,7 ;hacer mathreg2 positivo
    call   comp2
    btfsc  MATHREG3,7 ;hacer mathreg3 positivo
    call   comp3
    ;
    movlw  .33          ;33 shifts
    movwf  TEMP8
div_10:   call  shift_left1 ;shift left cociente 1 posicion
    bsf    MATHREG1+3,0 ;and lsb=1
    ;
    bcf    MATHFLAG,0 ;subtract divider de residuo
    subc   MATHREG2+3, MATHREG3+3
    subc   MATHREG2+2, MATHREG3+2
    subc   MATHREG2+1, MATHREG3+1
    subc   MATHREG2, MATHREG3
    btfss  MATHFLAG,0 ;check si residuo negativo
    goto   div_20
    bcf    MATHREG1+3,0 ;if residuo -, clear lsb cociente
    bcf    MATHFLAG,0 ; and add back divider to residuo
    addc   MATHREG2+3, MATHREG3+3
    addc   MATHREG2+2, MATHREG3+2
    addc   MATHREG2+1, MATHREG3+1
    addc   MATHREG2, MATHREG3
    ;
div_20:   decf  TEMP8,F ;decremento contador
    btfsc  STATUS,Z
    goto   div_30
    call   shift_left3 ;shift left residuo 1 posicion
    goto   div_10
    ;
div_30:   btfsc MATHSIGN,0
    call   comp1
    return

;*****Rutinas miscelaneas*****
mov3to4:movf    MATHREG3+4,W
    movwf  MATHREG4
    movf  MATHREG3+5,W
    movwf  MATHREG4+1
    movf  MATHREG3+6,W
    movwf  MATHREG4+2
    movf  MATHREG3+7,W
    movwf  MATHREG4+3
    return
;
mov4to2:movf    MATHREG4,W
    movwf  MATHREG2
    movf  MATHREG4+1,W
    movwf  MATHREG2+1
    movf  MATHREG4+2,W
    movwf  MATHREG2+2
    movf  MATHREG4+3,W
    movwf  MATHREG1+3
    return
;
;
shift_left1:
    bcf    STATUS,C
    rlf    MATHREG1+3,F

```

```

    rlf  MATHREG1+2,F
    rlf  MATHREG1+1,F
    rlf  MATHREG1,F
    return
;
shift_left2:
    bcf  STATUS,C
    rlf  MATHREG2+3,F
    rlf  MATHREG2+2,F
    rlf  MATHREG2+1,F
    rlf  MATHREG2,F
    return
;
shift_left3:
    bcf  STATUS,C
    rlf  MATHREG3+7,F
    rlf  MATHREG3+6,F
    rlf  MATHREG3+5,F
    rlf  MATHREG3+4,F
    rlf  MATHREG3+3,F
    rlf  MATHREG3+2,F
    rlf  MATHREG3+1,F
    rlf  MATHREG3,F
    return
;
clear3:    clrf  MATHREG3+7
    clrf  MATHREG3+6
    clrf  MATHREG3+5
    clrf  MATHREG3+4
    clrf  MATHREG3+3
    clrf  MATHREG3+2
    clrf  MATHREG3+1
    clrf  MATHREG3
    return
;
comp1:    comf  MATHREG1+3,F      ;complemento a 1
    comf  MATHREG1+2,F
    comf  MATHREG1+1,F
    comf  MATHREG1,F
    ;
    incf  MATHREG1+3,F      ; mas 1 para comp2
    btfss STATUS,Z
    goto  comp1_x
    incf  MATHREG1+2,F
    btfss STATUS,Z
    goto  comp1_x
    incf  MATHREG1+1,F
    btfss STATUS,Z
    goto  comp1_x
    incf  MATHREG1,F
comp1_x:return
;
comp2:    comf  MATHREG2+3,F      ;comp 1
    comf  MATHREG2+2,F
    comf  MATHREG2+1,F
    comf  MATHREG2,F
    ;
    incf  MATHREG2+3,F      ;mas 1 para comp2
    btfss STATUS,Z
    goto  comp2_x
    incf  MATHREG2+2,F

```

```

    btfss STATUS,Z
    goto comp2_x
    incf MATHREG2+1,F
    btfss STATUS,Z
    goto comp2_x
    incf MATHREG2,F
comp2_x:return
;
comp3:    comf MATHREG3+7,F    ;comp1
    comf MATHREG3+6,F
    comf MATHREG3+5,F
    comf MATHREG3+4,F
    comf MATHREG3+3,F
    comf MATHREG3+2,F
    comf MATHREG3+1,F
    comf MATHREG3,F
    ;
    movf FSR,W
    movwf TEMP7
    movlw MATHREG3+8
    movwf FSR
    movlw 08
    movwf TEMP8
comp_10:decf    FSR,F    ;mas 1 para comp2
    incf INDF,F
    btfss STATUS,Z
    goto comp_20
    decfsz    TEMP8,F
    goto comp_10
    ;
comp_20:movf    TEMP7,W
    movwf FSR
    return
;
end    ; FINAL

;*****FIN DEL PROGRAMA*****

```

ANEXO 3: PROGRAMAS EN TURBO C



Anexo A. Código de los programas en TurboC/C++

Recepción

```

#include <stdio.h>
#include <dos.h>
#include <conio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include "uart.h"
#include <fstream.h>

#define data_port      0x0378
#define status_port   data_port+1
#define control_port  data_port+2

int BuffHexRx[10],BuffBinRx[36];
unsigned char buffer_key[20];
int size_cadena,num,size_buffer,n_byte,ContChar,BytesInf,ContTimer;
unsigned char buffer_rx[50];int dato_rx,busy;
int cursorx,cursory; float lat,lon,latreal,longreal;
int char_val,char_inval,trama_val,trama_inval,var,flag,check_ok;
int port=0x3f8;
float LatRef,LongRef,cover;

/*****/
void LeeBusy (void)
{
busy=((inportb(status_port)^0x80)&0x80)>>7;
}
/*****/
void ConvDecBin(int num,int ini,int fin)
{
int i,j;
i=0;
do
{ BuffBinRx[fin-i]=num%2;
num=num/2;i++;
}while (num>=2);
BuffBinRx[fin-i]=num;
for(j=ini;j<fin-i;j++){
BuffBinRx[j]=0;}
}
/*****/
void LlenaBuffBinRx()
{int i;
for(i=0;i<=8;i++)
{
ConvDecBin(BuffHexRx[i],4*i,4*i+3);
}
for(i=0;i<=35;i++)printf("%d",BuffBinRx[i]);
}
/*****/
void LeeDato (void)
{
dato_rx= (inportb(status_port)&0x78)>>3;
flag=1;n_byte++;
gotoxy(50,1);printf("Byte:%d",n_byte-1);
}
/*****/

```

```

float ConvBinDec(int ini,int fin)
{ int x; float vall;
  vall=0;
  for(x=ini;x<=fin;x++)
  {
    vall=vall+ldexp(BuffBinRx[x],fin-x);
  }
  return vall;
}
/*****/
void LatLong(float latref,float longref)
{ double
declat,entlat,grad,vary,var4,var5,entgrad,declon,entlon,min,var1,var
r2,varx;

  cover=0.00195;
  lat=ConvBinDec(5,17);
  lon=ConvBinDec(19,31);
  declat=modf(latref/100,&entlat);
  latref=entlat*60+declat*100;
  declon=modf(longref/100,&entlon);
  longref=entlon*60+declon*100;
  if(BuffBinRx[4]<=0)lat=latref+lat*cover;else lat=latref-
lat*cover;
  if(BuffBinRx[18]<=0)lon=longref+lon*cover;else lon=longref-
lon*cover;
  varx=lat;
  var1=modf(varx/60,&var2);
  latreal=var2*100+fmod(lat,60);
  vary=lon;
  var4=modf(vary/60,&var5);
  longreal=var5*100+fmod(lon,60);
}
/*****/
void MuestraBuff(void)
{ int x; printf("\n");
  for(x=0;x<=8;x++)printf("%d ",BuffHexRx[x]);
  printf("\n");
  LlanaBuffBinRx();
  LatLong(LatRef,LongRef); printf("\n");
  printf("lat:%8.4f lon:%8.4f",latreal,longreal);
  return;
}
/*****/
void ControlaDato(void)
{ char check_hex;

check_hex=0x0f&(BuffHexRx[0]^BuffHexRx[1]^BuffHexRx[2]^BuffHexRx[3]
^BuffHexRx[4]^BuffHexRx[5]^BuffHexRx[6]^BuffHexRx[7]);
  if(check_hex==BuffHexRx[8])check_ok=1;
  else check_ok=0;
}
/*****/
void ReceiveData(void)
{
  int x,cont1,cont2,veces;      var=0;
  num=0;
  veces=5;
  LeeBusy();
  while( busy==1 )
  {

```

```

cont1=veces;
do
{
  LeeBusy();
  if (busy==1) { cont1--;} //CONTROL DE GLITCHES
  else {break;}
  delay(1); if (kbhit()) break;
}while(cont1!=0);
if (cont1==0)
{ LeeDato(); // DATO VALIDO!!!!
  do{
    LeeBusy();
    if(busy==0)
    { cont2=veces;
      do{ LeeBusy();
        if(busy==0){cont2--;}
        else{break;}
        delay(1);
        if (kbhit()) break;
      } while(cont2!=0);
    }
    if (kbhit()) break;
  }while(busy==1);
  break;
}
}
}
/*****/
void main ()
{ int val; trama_val=0;trama_inval=0;
  n_byte=0; BytesInf=9;
  clrscr();
  printf("LatRef:");scanf("%f",&LatRef);
  printf("LongRef:");scanf("%f",&LongRef);clrscr();
  printf("RECEPCION \n");

  while(!kbhit())
  {bucle1:
  flag=0;
  ReceiveData();
  // if(dato_rx!=12)goto bucle1;
  if(dato_rx=12&&flag!=0)
  {
    ContChar=BytesInf+1;
    do{ContTimer=0;
      do{ delay(1);
        ContTimer++;
        if(ContTimer>150)
        {if(n_byte>2)trama_inval++;
          n_byte=0;
          goto bucle1;
        }
        ReceiveData();
      }while(flag!=1);
    }if(flag!=0){
      BuffHexRx[n_byte-2]=dato_rx;
      ContChar--;
      flag=0;
    }

  }while(ContChar!=0);
}

```

```
if(ContChar<=0&&n_byte==10)
{ if(dato_rx=11)
  {ControlaDato();
   if(check_ok!=0)
    { trama_val++;
     MuestraBuff();
     /*******/
     ofstream f_out("c:\\foutput.dat");
     f_out<<latreal<<" "<<longreal<<"\n";
     f_out.close();
     /*******/
     n_byte=0;
    }
   else trama_inval++;
  }
 else trama_inval++;
}
gotoxy(50,2);
printf("TramInv:%d TramaVal:%d",trama_inval,trama_val);
}
}
if(kbhit()) return ;
return;
}
/****** Fin del programa de recepción *****/
```



Transmisión

```
//PROGRAMA DE TRANSMISION DE INFORMACION GPS
//POR MEDIO DE TONOS DTMF
//SE ENSAMBLA CON # de cabecera y * de cola
// # id lat lat lat lat/lon lon lon lon chk *

#include <conio.h>
#include <stdio.h>
#include <dos.h>
#include "uart.h"
#include <string.h>
#include <math.h>
#include <stdlib.h>

#define data_port      0x0378
#define status_port    data_port+1
#define control_port   data_port+2
#define cuentas 8192

int ContBuffSerial,BufferTx[11],BuffAssembler[36],c_insert,pos;
unsigned char BuffSerial[256],BuffLat[8],BuffLong[9];
float LongRef,LongDif,LongActual,LatRef,LatDif,LatActual;
int ok,time_ptt,time_st,present,absent,single_tone;
double cover,DeltaLat,DeltaLong; unsigned int i,tramas; int
checksum;
int buffer_key[10][4],val_bk[9]; unsigned char dato; int
flag,port;
unsigned char head[5]="GPGGA" ;

/*****
*/
void UART_init(int port) {
    /*Set formato del dato y data rate*/
    outportb(port+LCR, 0x83); /* 8-n-1 and DLAB = 1 */
    outportb(port+DL_L, 0xC); /* 9600 bps */
    outportb(port+DL_H, 0);
    outportb(port+LCR, 0x3); /* reset DLAB = 0 */
}
/*****
*/
unsigned char UART_getchar(int port)
{
    /* sondea el puerto hasta que el dato este disponible y lo lee */
    while (((inportb(port+LSR )) & RDR ) == 0);
    if(kbhit()) return 0;
    return inportb(port);
}
/*****
*/
void RecibeSerialPort( unsigned port )
{
    dato=UART_getchar(port);
    if(dato=='$')ContBuffSerial=0;
    BuffSerial[ContBuffSerial]=dato;
    //printf("%c",BuffSerial[ContBuffSerial]);
    ContBuffSerial++;
    if(kbhit())return ;
    return ;
}
}
```

```

/*****
*/
//CONVIERTE A NUM1 EN BINARIO Y LO PONE EN BuffAssembler
//DESDE LA POSICION INI HASTA FIN
void ConvDecBin(float num1,int ini,int fin)
{
  int i,j,num;
  i=0;num=fabs(num1);
  do
  { BuffAssembler[fin-i]=num%2;
    num=num/2;i++;
  }while (num>=2);
  BuffAssembler[fin-i]=num;
  for(j=ini+1;j<fin-i;j++){
    BuffAssembler[j]=0;}
}
/*****
*/
void ProcesaDatos ()
{
  double LatEnt,LongEnt;
  float id,eof;
  double LatDec,LongDec;
  double DecLongRef,EntLongRef;
  double DecLongActual,EntLongActual;
  double grad1,grad2;
  double DecLatRef,EntLatRef;
  double DecLatActual,EntLatActual;
  double minref,minact,minref2,minact2;
  double xLatRef,xLatActual,xLongRef,xLongActual;

  cover=0.00195;
  id=1;
  gotoxy(1,5);
  /*****/
  LatActual=atof(BuffLat);
  LongActual=atof(BuffLong);
  printf("\nLat:%f",LatActual);
  printf("\nLong:%f\n",LongActual);

  xLatRef=LatRef/100;
  DecLatRef=modf(xLatRef,&EntLatRef);

  xLatActual=LatActual/100;
  DecLatActual=modf(xLatActual,&EntLatActual);

  minref=EntLatRef*60+DecLatRef*100;
  minact=EntLatActual*60+DecLatActual*100;

  LatDif=minact-minref;

  xLongRef=LongRef/100;
  DecLongRef=modf(xLongRef,&EntLongRef);

  xLongActual=LongActual/100;
  DecLongActual=modf(xLongActual,&EntLongActual);

  minref2=EntLongRef*60+DecLongRef*100;
  minact2=EntLongActual*60+DecLongActual*100;

  LongDif=minact2-minref2;

```

```

    if(LongDif<0) BuffAssembler[18]=1;
    else BuffAssembler[18]=0;

    if(LatDif<0) BuffAssembler[4]=1;
    else BuffAssembler[4]=0;

    DeltaLat=fabs(LatDif/cover);
    LatDec=modf(DeltaLat,&LatEnt);

    DeltaLong=fabs(LongDif/cover);
    LongDec=modf(DeltaLong,&LongEnt);

    if(LatDec>=0.5) DeltaLat=LatEnt+1; else DeltaLat=LatEnt;
    if(LongDec>=0.5) DeltaLong=LongEnt+1; else DeltaLong=LongEnt;
    /*****
    ConvDecBin(id,0,3);
    ConvDecBin(DeltaLat,4,17);
    ConvDecBin(DeltaLong,18,31);
    */
}
/*****
*/
void LlenaBufferTx(void);

int ConvBinDec(int n_byte)
{ int x,vall;
  x=n_byte;

  vall=(buffer_key[x][0])*8+(buffer_key[x][1])*4+(buffer_key[x][2])*2
    +(buffer_key[x][3]);
  return vall;
}
/*****
*/
void BuffAssemblerToBuff_key()
{ int x,y;
  for(y=0;y<=7;y++)
  {
    for(x=0;x<=3;x++)
    {
      buffer_key[y][x]= BuffAssembler[x+4*(y)];
    }
  }
}
/*****
*/
void BuffKeyToDec() //convierte todo el buff_key en decimal
{ int z;
  for(z=0;z<=7;z++)
  {
    val_bk[z]=ConvBinDec(z);
  }
}
/*****
*/
void PoneChecksum(void)
{int n;
  checksum=0x0f&(val_bk[0]^val_bk[1]^val_bk[2]^val_bk[3]
    ^val_bk[4]^val_bk[5]^val_bk[6]^val_bk[7]);
}

```

```

ConvDecBin(checksum,31,35);
val_bk[8]=checksum;
for(n=0;n<=3;n++)
  { buffer_key[8][n]= BuffAssembler[n+32];}
}
/*****
*/
void EnsamblaBufferTx()
{ int x,y,z,end;
  ProcesaDatos();
  BuffAssemblerToBuff_key();
  BuffKeyToDec();
  PoneChecksum();
  LlenaBufferTx(); printf("\n");
  for(z=0;z<=10;z++)printf("%x ",BufferTx[z]);
  printf("\n");
  for(y=0;y<=8;y++)
  {
    for(x=0;x<=3;x++)
    {
      printf("%d",buffer_key[y][x]);
    }
    printf(" ");
  }
}
/*****
*/
void ChequeaBuffer()
{ int i,j;
i=0;
do{
  if (BuffSerial[i]=='$')
  {
    for(j=0;j<=4;j++)
    {
      if(BuffSerial[i+1+j]==head[j]) {ok=1;}
      else ok=0;
    }
    if(ok==1)
    {if(BuffSerial[i+6]==BuffSerial[i+13])
      for(j=0;j<=7;j++)
      { BuffLat[j]=BuffSerial[i+14+j];}
      for(j=0;j<=8;j++)
      {BuffLong[j]=BuffSerial[i+25+j];}
    }
  }
  if(ok==0) {return;}
  else i++;

}while(i<=ContBuffSerial);
return;
}
/*****
*/
void DTMF_init()
{
  time_ptt=100;
  time_st=200;
  present=50;absent=50;
  single_tone=0xd0; //tono de 1477 Hz
  //clrscr();

```

```

printf("TRANSMISION\n");
printf("TimeTono:%dms      \n",present);
}
/*****
*/
void LlenaBufferTx(void)
{
    int y,end ;
    BufferTx[0]=0xde;
    BufferTx[10]=0x7e;
    for(y=1;y<=9;y++)
    {
        switch(val_bk[y-1]){
            case 1 : BufferTx[y]=0x77;break;
            case 2 : BufferTx[y]=0xb7;break;
            case 3 : BufferTx[y]=0xd7;break;
            case 4 : BufferTx[y]=0x7b;break;
            case 5 : BufferTx[y]=0xbb;break;
            case 6 : BufferTx[y]=0xdb;break;
            case 7 : BufferTx[y]=0x7d;break;
            case 8 : BufferTx[y]=0xbd;break;
            case 9 : BufferTx[y]=0xdd;break;
            case 10 : BufferTx[y]=0xbe;break;
            case 11 : BufferTx[y]=0x7e;break;
            case 12 : BufferTx[y]=0xde;break;
            case 13 : BufferTx[y]=0xe7;break;
            case 14 : BufferTx[y]=0xeb;break;
            case 15 : BufferTx[y]=0xed;break;
            case 0 : BufferTx[y]=0xee;break;
            default : BufferTx[y]=0;break;};
    }
}
/*****
*/
void SendInformation(void)
{
    int point=0;
    tramas++;
    outportb(control_port,0x05^0x01);
    delay(time_ptt);//ptt + activo 5089
    outportb(data_port,single_tone);//Envio single tone
    delay(time_st);
    do{
        outportb(data_port,BufferTx[point]);
        i++;
        gotoxy(1,3);printf("# de char Tx: %u      #tramas Tx:
%u\n",i,tramas);
        delay(present);
        outportb(data_port,0xFF);
        delay(absent);
        point++;      if(kbhit()) break;
    }while(point<10);
    outportb(control_port,0x00^0x01); //desactivo ptt y tono
}
/*****
*/
void main()
{
    int x; tramas=0; i=0;
    clrscr();
    port=0x3f8;

```

```
printf("LatRef :");scanf("%f",&LatRef);
printf("LongRef:");scanf("%f",&LongRef);
UART_init(port);
DTMF_init();
while(!kbhit())
{
    ContBuffSerial=0;
    do{
        RecibeSerialPort(port);
        if(kbhit()) return ;
        }while(dato!='*');
    ChequeaBuffer();
    EnsamblaBufferTx();
    SendInformation();

}
return ;
}
/***** Fin del programa de Transmisión *****/
```





GLOSARIO

SQUELCH O SILENCIADOR: Valor ajustable para silenciar un receptor si la energía de RF recibida cae por debajo de un cierto valor del umbral.

DTMF: Tonos duales multifrecuencia. Se refiere a la señal compuesta de dos tonos audibles.

VHF: Frecuencia súper alta que va de los 30 a los 300 MHz.

SINAD: Relación de señal a ruido (SNR) mas distorsión.

WAYPOINT: Se refiere a una ubicación específica marcada en un plano, mapa o ruta determinada.

DATUM: Descripción matemática de la Tierra. Típicamente se define como un elipsoide de revolución, con sus dos radios (ecuatorial y polar), o su radio ecuatorial y el nivel de "achatamiento".

Hay que tener en cuenta que diferentes zonas del mundo y diferentes mapas pueden utilizar un "datum" distinto, por lo que pueden no encajar exactamente. Se trata de un detalle a tener muy en cuenta.

El "datum" actual y más moderno es el WGS-84

WAAS: Sistema de incremento de precisión de un área amplia.

DGPS: Sistema de Posicionamiento Global Diferencial.

SELECT AVAILABILITY: Mecanismo por medio del cual EE.UU. introduce errores en las referencias enviadas por los satélites, con el fin de disminuir la precisión del sistema. Actualmente está desactivado.



ANEXO 5: DIAGRAMA ESQUEMATICO

