

# **ESCUELA POLITÉCNICA NACIONAL**

**FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**

**“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA PROTOTIPO DE LOCALIZACIÓN VEHICULAR EN LA BANDA VHF CON TÉCNICAS DE MODULACIÓN DIGITAL EN TIEMPO REAL UTILIZANDO EL MÓDULO DE DESARROLLO DEL MICROPROCESADOR ADSP-2181”**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES**

**CHRISTIAN PATRICIO ACEVEDO PÁEZ**

**[christian\\_acevedo@hotmail.com](mailto:christian_acevedo@hotmail.com)**

**MARCO RENATO MEJÍA RUIZ**

**[marcomr01@yahoo.com](mailto:marcomr01@yahoo.com)**

**DIRECTOR: ING. FERNANDO VÁSQUEZ**

**[fvasquez@watching.com.ec](mailto:fvasquez@watching.com.ec)**

**Quito, Noviembre del 2009**

## DECLARACIÓN

Nosotros, Christian Patricio Acevedo Páez y Marco Renato Mejía Ruiz, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

---

Christian P. Acevedo P.

---

Marco R. Mejía R.

## **CERTIFICACIÓN**

Certifico que el presente trabajo fue desarrollado por Christian Patricio Acevedo Páez y Marco Renato Mejía Ruiz, bajo mi supervisión.

---

Ing. Fernando Vásquez  
DIRECTOR DEL PROYECTO

## **AGRADECIMIENTO**

Agradezco primeramente a Dios por habernos dado la bendición de la vida, a mis padres que fueron y son el apoyo más importante en todas las etapas de mi vida y que han logrado que culmine con éxito el presente proyecto de titulación, a mis hermanos con los que imperecederamente conté con su ayuda incondicional, a mis abuelitos que constantemente estuvieron allí y que influenciaron de manera positiva en gran parte de mi existencia, a todos mis tíos y tías que fueron un aliento para continuar y concluir con satisfacción este ciclo de mi vida, a todos mis primos con los que siempre compartí gratas experiencias, a todos mis amigos y amigas que tuve el honor de conocerlos y un especial agradecimiento a nuestro director de tesis y a todos los ingenieros que fueron un soporte indispensable para el cumplimiento del presente proyecto.

Christian Patricio Acevedo Páez

## **DEDICATORIA**

Todo el esfuerzo puesto en este trabajo lo dedico a mi abuelita “Mamita Blanqui” que fue ejemplo de sinceridad, inteligencia, humildad, generosidad, cariño, don de gente, entre otras y que siempre supo estar allí cuando la necesitaba, nunca te olvidaré y gracias por todo.

Christian Patricio Acevedo Páez

## **AGRADECIMIENTO**

A Dios por permitir que finalice con éxito una etapa más de mi vida, a mi linda familia a quien me debo por completo, a todos quienes considero mis verdaderos e invaluable amigos. De manera especial a nuestro director de tesis Ing. Fernando Vásquez y a todos quienes forman parte de las empresas *Watching* y *MCF* gracias por todo el inmejorable apoyo recibido.

Marco Renato Mejía Ruiz

## **DEDICATORIA**

Con muchísimo amor a mi madre Rosita, a mi padre Luis, a mis hermanitos Mauricio Roberto, Milton Raúl, Marlon Ramiro y Marcelo Robinson, y a mis hermanitas Margoth Rocío y Marylin Ruth. Los amo y llevo en mi corazón.

Marco Renato Mejía Ruiz

## INTRODUCCIÓN

A lo largo de su historia, el Ecuador se ha caracterizado por ser tecnológicamente dependiente, limitándose a satisfacer sus necesidades de progreso mediante la adquisición de tecnología desarrollada fuera de sus fronteras, y es que quizás no existan las condiciones apropiadas para por lo menos satisfacer la demanda interna, peor aún perfilarnos como potenciales exportadores, pero si somos capaces de sentar las bases de una cultura emprendedora no dependiente, en la que se desarrollen nuevas aplicaciones, en base a una previa asimilación de conocimientos, o se construyan otras existentes en las que los recursos de los que se dispone para tal efecto sean eficientemente aprovechados, re potenciando el uso de tecnología disponible y de su infraestructura instalada.

La coyuntura es más favorable que en otras épocas; la evolución fantástica de las tecnologías de la información, el auge del mercado de servicios y su mutua complementación, ha propiciado un avance espectacular en cuanto a la utilidad y aplicabilidad de ciertas tecnologías de la ingeniería en cualquier ámbito del desenvolvimiento humano; añadiendo a esto una política de apoyo que incentiva el consumo de productos nacionales.

Son estas nuestras convicciones e incentivos, lo que nos han llevado al planteamiento de un problema en particular, desarrollar una plataforma para la transmisión y recepción de información de navegación sobre otra que solamente soporta voz, dicho problema es solucionado en el presente proyecto de titulación, cuyo producto final es un sistema básico de localización vehicular vía radio en la banda VHF con amplias expectativas de sobrepasar el nivel académico y convertirse en una aplicación comercial.

## RESUMEN

La tecnología cada vez potencia más la prestación de nuevos y mejorados servicios, que permiten la satisfacción tanto del ofertante como del demandante, de tal modo que las empresas que lucran de esta actividad han de mantenerse al tanto para no ceder terreno en el campo de la competencia, desde este punto de vista la Localización Automática Vehicular permite obtener un plus a quienes la implementan, ya que a partir de ello se puede garantizar la prestación del servicio ante el cliente y aun más obtener beneficios económicos por ahorro debido a un adecuado control logístico que facilita la planificación y gestión de actividades. Por estas cualidades esenciales y otras más, es creciente la tendencia de adquirir un sistema que permita gestionar los activos móviles de una empresa, lo cual implica poseer un sistema de comunicaciones operativo para manejar el envío y recepción de información.

Pero, ¿Qué sucede cuando se tiene instalada y operativa una infraestructura de comunicaciones?, es posible que los sistemas de localización que se ofrecen en el mercado se acoplen perfectamente o simplemente haya que rediseñar, reemplazar totalmente dicha infraestructura por no prestar las condiciones para la transmisión – recepción de datos, o inclusive instalar una paralela, tal es el caso de redes de comunicaciones vía radio que actualmente no tienen cabida en la aplicación mencionada.

Planteamos solucionar el problema de tener que reemplazar toda una infraestructura de radio comunicación VHF - UHF o instalar una paralela a esta, mediante el desarrollo de una plataforma para la transmisión de información de navegación que se adapte satisfactoriamente a la mencionada red que solo permite el intercambio de mensajes de voz, de manera que se mantenga intacta esta funcionalidad.

Para tales efectos se usaron herramientas disponibles, los datos de navegación originados en un sistema de posicionamiento global GPS deben ser capturados por la unidad móvil y enviados a una central para lo cual es necesario adaptar esta información al canal inalámbrico de comunicaciones en la banda VHF, esto es básicamente implementar la plataforma de datos, todo permitido por un sistema de

Procesamiento Digital de Señal (DSP) representado en el EZ Kit Lite ADSP 2181. La programación de la funcionalidad del sistema fue realizada en *Assembler* mediante el software *VisualDSP 3.5* que permite el manejo de microcontroladores DSP ADI<sup>1</sup>, dicha funcionalidad se basa esencialmente en la inclusión de estructuras DSP como generación de ondas y filtros digitales, para lo cual también se echó mano de un software de tendencia actual *MatLab*. El acoplamiento físico entre los elementos de transmisión – recepción (radios VHF) y la plataforma de datos (Ez Kit Lite) se la realiza mediante circuitos analógicos diseñados y construidos como parte de la implementación del sistema básico de Localización Automático Vehicular, permitiendo de esta manera la coexistencia de funcionalidades, transmisión de voz y de información de navegación.

La implementación de la plataforma para la transmisión – recepción de información de navegación obtenida de un GPS se la realiza mediante generación y modulación de una portadora en BPSK, técnica obtenida de un análisis costo beneficio entre los esquemas típicos de radio digital, tomando en consideración velocidad de transmisión necesaria, susceptibilidad al ruido y facilidad de implementación entre las principales.

El instante adecuado para la transmisión de información de navegación es evaluado mediante un módulo que provee la suficiente inteligencia al sistema para tomar en consideración parámetros como movimiento vehicular y estado de pánico de modo que no exista interferencia entre la funcionalidad primitiva de voz y la de transmisión de información de navegación.

Los resultados obtenidos son completamente satisfactorios y son visualizados en tiempo real en un sistema de información geográfica de tendencia actual, *Google Earth*, y adicionalmente almacenando la información recibida en una base de datos en *Microsoft Access* posibilitando un análisis posterior de la misma.

---

<sup>1</sup> Analog Devices Inc.

## PRESENTACIÓN

El presente documento describe todas las consideraciones necesarias y suficientes, tanto teóricas como prácticas, así como los diversos procesos que se han efectuado para lograr concretar el objetivo primordial de esta tesis, implementar un prototipo para localización vehicular vía radio VHF, usando un dispositivo GPS y efectuando los procesos necesarios en un sistema de procesamiento digital de señal, utilizando el Ez Kit Lite ADSP 2181, cuyo elemento fundamental es un microcontrolador DSP.

Para este efecto se ha dividido la presente redacción en cinco capítulos:

El capítulo uno es una descripción del contexto actual de la localización vehicular y de las herramientas que usa para lograr su objetivo, realizando así un realce de su vigente importancia, permitiendo además conocer de cerca las diversas tendencias tecnológicas características de su implementación. Se hace también una breve síntesis de la principal herramienta cognoscitiva, el procesamiento digital de señal, y a sus principales aplicaciones a utilizarse en el presente proyecto de titulación, realizando además una descripción del instrumento disponible que lo hace posible, el EZ Kit Lite ADSP 2181.

El capítulo dos realiza un análisis de todos aquellos elementos que necesariamente se deben tomar en cuenta para la implementación de cada una de las partes constitutivas del sistema que hará posible el cumplimiento del objetivo principal, además presenta los lineamientos que deben seguirse en cada una de estas partes de modo que la implementación se lleve a cabo de una manera guiada.

El Capítulo tres describe la implementación del proyecto, en tres áreas fundamentales: Firmware, Hardware y Software, el primero de ellos destinado a establecer la plataforma más adecuada para la transmisión de información de navegación, así como de manejar la inteligencia en el envío de dichos datos; el segundo con la tarea de establecer la interfaz física entre los elementos empleados y facilitar la integración de la nueva funcionalidad; finalmente el software que permite

mostrar de manera inteligible la información de navegación al usuario mediante un mapa digitalizado.

El capítulo cuatro recopila todas aquellas pruebas, tanto de laboratorio como de campo, efectuadas en las diversas fases de desarrollo del presente proyecto, estableciendo conclusiones que llevaron a incluir mejoras en el funcionamiento de los prototipos y en los módulos de programa que componen el firmware.

El capítulo cinco finalmente hace referencia a las conclusiones y recomendaciones rescatadas de la realización del proyecto de titulación.

## CONTENIDO

INTRODUCCIÓN .....	VIII
RESUMEN.....	IX
PRESENTACIÓN.....	XI
 <b>CAPITULO 1</b>	
<b>CONCEPTOS GENERALES</b>	
<b>1.1. LOCALIZACIÓN AUTOMÁTICA VEHICULAR (LAV).....</b>	<b>1</b>
1.1.1. DESCRIPCIÓN GENERAL.....	1
1.1.2. SISTEMAS LAV ACTUALES .....	2
1.1.2.1. Tipos de Sistemas LAV .....	3
1.1.2.1.1. Según el origen de los datos de posicionamiento .....	3
1.1.2.1.1.1. Sistemas basados en GPS .....	3
1.1.2.1.1.2. Sistemas no basados en GPS.....	4
1.1.2.1.1.3. Combinación de sistemas GPS y no GPS .....	6
1.1.2.1.2. Según la característica de tiempo real .....	7
1.1.2.1.2.1. Sistemas en línea .....	7
1.1.2.1.2.2. Sistemas fuera de línea .....	8
1.1.2.1.3. Según la red que los soporta .....	8
1.1.2.1.3.1. Sistemas sobre red de datos celular .....	8
1.1.2.1.3.2. Sistemas vía radiofrecuencia .....	10
1.1.2.1.3.3. Sistemas satelitales .....	12
<b>1.2. SISTEMAS DE INFORMACIÓN GEOGRÁFICA (SIG) .....</b>	<b>13</b>
1.2.1. IMPORTANCIA ACTUAL .....	13
1.2.2. FUNCIONAMIENTO DE UN SIG.....	14
1.2.3. TÉCNICAS UTILIZADAS EN LOS SISTEMAS DE INFORMACIÓN GEOGRÁFICA .....	15
1.2.3.1. Creación y obtención de datos.....	15
1.2.3.2. Representación de datos .....	15
1.2.3.2.1. Raster .....	15
1.2.3.2.2. Vectorial.....	16
1.2.4. SOFTWARE SIG.....	18
1.2.4.1. Google Earth.....	18
<b>1.3. PROCESAMIENTO DIGITAL DE SEÑAL.....</b>	<b>20</b>
1.3.1. SISTEMAS DE PROCESAMIENTO DIGITAL DE SEÑAL .....	21
1.3.1.1. Esquema Básico .....	21
1.3.1.2. Microcontrolador DSP.....	23
1.3.2. FILTROS DIGITALES.....	27
1.3.2.1. Técnicas de Filtrado digital .....	27
1.3.2.1.1. Filtros FIR e IIR.....	27
1.3.2.1.1.1. IIR.....	28
1.3.2.1.1.2. FIR.....	28

1.3.2.2.	Diseño y simulación utilizando <i>Matlab</i> .....	29
1.3.2.2.1.	<i>Ejemplo de realización de un filtro digital con FDATAOL</i> .....	32
1.3.3.	GENERACIÓN DE FORMAS DE ONDA .....	35
<b>1.4.</b>	<b>EZ KIT LITE ADSP – 2181 .....</b>	<b>41</b>
1.4.1.	DESCRIPCIÓN GENERAL .....	41
1.4.2.	ARQUITECTURA BASE .....	42
1.4.2.1.	Microcontrolador ADSP 2181 .....	42
1.4.2.1.1.	<i>Unidades Computacionales</i> .....	42
1.4.2.1.1.1.	<i>Unidad aritmética Lógica (ALU)</i> .....	43
1.4.2.1.1.2.	<i>Unidad de Multiplicación y Acumulación (MAC)</i> .....	43
1.4.2.1.1.3.	<i>Unidad de desplazamiento (SHIFTER)</i> .....	44
1.4.2.1.2.	<i>Control del programa</i> .....	44
1.4.2.1.2.1.	<i>Secuenciador de Programa</i> .....	44
1.4.2.1.2.2.	<i>Interrupciones</i> .....	45
1.4.2.1.3.	<i>Transferencia y almacenamiento de datos</i> .....	46
1.4.2.1.3.1.	<i>Tipos de Buses</i> .....	46
1.4.2.1.3.2.	<i>Generadores de direcciones de datos</i> .....	46
1.4.2.1.3.3.	<i>Memoria</i> .....	47
1.4.2.2.	Periféricos Incorporados .....	47
1.4.2.2.1.	<i>Puerto Serial</i> .....	47
1.4.2.2.2.	<i>Banderas entrada/salida</i> .....	48
1.4.2.3.	Códec AD1847 .....	48
1.4.3.	SOFTWARE DE DESARROLLO .....	49
1.4.3.1.	Breve descripción del tipo de programación utilizado .....	49
1.4.3.2.	VisualDSP++ 3.5 .....	49
1.4.3.2.1.	<i>Puesta en marcha de Visual DSP++ 3.5</i> .....	51
1.4.3.2.2.	<i>Ejemplo práctico de Utilización de VisualDSP++ 3.5</i> .....	54
1.4.4.	PROGRAMA HOST EZ KIT LITE .....	63
<b>1.5.</b>	<b>TRANSMISIÓN DE VOZ MEDIANTE RADIOS VHF .....</b>	<b>65</b>
1.5.1.	DESCRIPCIÓN BÁSICA .....	65
1.5.2.	MODOS DE OPERACIÓN .....	66
1.5.2.1.	Modo Simplex .....	66
1.5.2.2.	Modo Semi – dúplex .....	67
1.5.3.	DESCRIPCIÓN DEL EQUIPO EMPLEADO EN COMUNICACIONES VÍA RADIO .....	67
1.5.3.1.	Modo de manipulación .....	68

## CAPÍTULO 2

### CONSIDERACIONES DE DISEÑO

<b>2.1.</b>	<b>CONSIDERACIONES EN EL TRANSMISOR.....</b>	<b>70</b>
2.1.1.	RECEPCIÓN GPS .....	71
2.1.1.1.	Aspectos de Interfaz del dispositivo GPS .....	72
2.1.1.2.	La Trama NMEA GPRMC .....	73
2.1.1.2.1.	<i>Información proporcionada</i> .....	74

2.1.1.3.	Trama de información a transmitir en el sistema básico LAV.....	75
2.1.2.	TRANSMISIÓN EN VHF .....	76
2.1.2.1.	Ancho de banda del canal disponible.....	76
2.1.2.2.	Frecuencias portadoras disponibles en modo semi – dúplex.....	78
2.1.3.	ACOPLAMIENTO FÍSICO DE VOZ Y DATOS EN TRANSMISIÓN.....	78
2.1.4.	PROCESAMIENTO DIGITAL DE SEÑAL EN TRANSMISIÓN.....	80
2.1.4.1.	Módulo de Generación de onda modulada .....	82
2.1.4.1.1.	Modulación ASK.....	86
2.1.4.1.2.	Modulación FSK.....	88
2.1.4.1.3.	Modulación BPSK.....	90
2.1.4.2.	Módulo de recepción y almacenamiento de carácter desde el GPS .....	93
2.1.4.3.	Módulo de detección y almacenamiento de la trama NMEA GPRMC .....	93
2.1.4.4.	Módulo de edición de la trama almacenada.....	94
2.1.4.5.	Módulo de evaluación de apertura automática del canal .....	94
2.2.	CONSIDERACIONES EN EL RECEPTOR .....	96
2.2.1.	ACOPLAMIENTO FÍSICO DE VOZ Y DATOS EN RECEPCIÓN .....	97
2.2.2.	PROCESAMIENTO DIGITAL DE SEÑAL EN RECEPCIÓN .....	98
2.2.2.1.	Módulo de Filtrado .....	100
2.2.2.2.	Módulo de demodulación.....	100
2.2.2.3.	Módulo de discriminación de bits .....	100
2.2.2.4.	Módulo de validación de la información recibida .....	101
2.2.2.5.	Módulo de transmisión serial .....	101
2.2.3.	CONSIDERACIONES DE SOFTWARE.....	101
2.3.	ESTRUCTURA DE DATOS A USAR.....	102
2.3.1.	MODO DE TRANSMISIÓN ASÍNCRONO.....	102
2.3.2.	ESTRUCTURA DE CARACTER .....	103

## CAPÍTULO 3

### IMPLEMENTACIÓN DEL PROTOTIPO

3.1.	FIRMWARE .....	104
3.1.1.	TRANSMISOR.....	106
3.1.1.1.	Módulo de recepción y almacenamiento de caracter desde el GPS .....	107
3.1.1.2.	Módulo de detección y almacenamiento trama NMEA GPRMC .....	111
3.1.1.3.	Módulo de edición de trama almacenada.....	114
3.1.1.4.	Módulo de evaluación de apertura automática de canal .....	117
3.1.1.5.	Módulo de generación de onda modulada.....	121
3.1.1.5.1.	Modulación ASK.....	122
3.1.1.5.2.	Modulación FSK.....	129
3.1.1.5.3.	Modulación BPSK.....	131
3.1.2.	RECEPTOR.....	133
3.1.2.1.	Módulo de filtrado.....	134
3.1.2.2.	Módulo de demodulación.....	137
3.1.2.2.1.	Demodulación ASK .....	137

3.1.2.2.2.	<i>Demodulación FSK</i> .....	140
3.1.2.2.3.	<i>Demodulación BPSK</i> .....	143
3.1.2.3.	Módulo de discriminación de bits .....	146
3.1.2.4.	Módulo de validación de la información recibida .....	150
3.1.2.5.	Módulo de transmisión serial .....	152
3.1.3.	TIPO DE MODULACIÓN DEFINITIVA A UTILIZAR .....	155
3.1.4.	IMPLEMENTACIÓN FINAL DEL FIRMWARE .....	157
<b>3.2.</b>	<b>SOFTWARE</b> .....	<b>159</b>
3.2.1.	MÓDULO DE TRATAMIENTO DE DATOS .....	159
3.2.2.	VISUALIZACIÓN DE RESULTADOS EN GOOGLE EARTH .....	160
3.2.3.	ALMACENAMIENTO DE INFORMACIÓN RECIBIDA EN BASE DE DATOS .....	161
<b>3.3.</b>	<b>HARDWARE (ACOPLAMIENTO FÍSICO DE ELEMENTOS).....</b>	<b>162</b>
3.3.1.	INTERFAZ TARJETA DE DESARROLLO – RADIO BASE .....	163
3.3.1.1.	Esquemático del circuito .....	163
3.3.1.2.	Operación del circuito .....	164
3.3.2.	INTERFAZ DISPOSITIVO GPS – TARJETA DE DESARROLLO .....	166
3.3.3.	CONTROL AUTOMÁTICO DE GANANCIA .....	167
3.3.3.1.	Esquemático del circuito .....	167
3.3.3.2.	Operación del circuito .....	168
<b>3.4.</b>	<b>CONSTRUCCIÓN FINAL DE LOS PROTOTIPOS</b> .....	<b>171</b>
3.4.1.	TRANSMISOR .....	171
3.4.2.	RECEPTOR .....	174
<b>3.5.</b>	<b>ESTABLECIMIENTO DE LA SECCIÓN REPETIDORA</b> .....	<b>177</b>
 <b>CAPÍTULO 4</b>		
<b>REALIZACIÓN DE PRUEBAS Y ANÁLISIS DE RESULTADOS</b>		
<b>4.1.</b>	<b>VERSIONES DE EVALUACIÓN DE LOS PROTOTIPOS</b> .....	<b>180</b>
4.1.1.	PRIMERA VERSIÓN DE LOS PROTOTIPOS .....	180
4.1.2.	SEGUNDA VERSIÓN DE LOS PROTOTIPOS .....	184
4.1.3.	TERCERA VERSIÓN DEL PROTOTIPO TRANSMISOR .....	185
<b>4.2.</b>	<b>TIPOS DE PRUEBAS REALIZADAS</b> .....	<b>186</b>
4.2.1.	PRUEBAS DE LABORATORIO .....	187
4.2.1.1.	Pruebas a los módulos de procesamiento digital señal .....	187
4.2.1.1.1.	<i>Pruebas a los esquemas de modulación</i> .....	187
4.2.1.1.1.1.	<i>Modulación y Demodulación ASK</i> .....	188
4.2.1.1.1.2.	<i>Modulación y Demodulación FSK</i> .....	190
4.2.1.1.1.3.	<i>Modulación y Demodulación PSK</i> .....	192
4.2.1.1.2.	<i>Pruebas a la sección de discriminación y transmisión serial</i> .....	195
4.2.1.1.3.	<i>Pruebas al módulo de Apertura Automática del Canal</i> .....	197

4.2.1.2.	Pruebas al hardware de acoplamiento físico.....	200
4.2.1.2.1.	Control Automático de ganancia .....	200
4.2.2.	PRUEBAS DE CAMPO EN MODO SEMI - DÚPLEX .....	202
4.2.2.1.	Primera Prueba .....	203
4.2.2.2.	Segunda Prueba.....	205
4.2.2.3.	Tercera Prueba.....	207
4.2.2.4.	Cuarta Prueba .....	210
4.2.2.5.	Quinta Prueba .....	213
4.2.2.6.	Sexta Prueba .....	215

## **CAPÍTULO 5**

### **CONCLUSIONES Y RECOMENDACIONES**

5.1.	CONCLUSIONES.....	II
5.2.	RECOMENDACIONES.....	V
	BIBLIOGRAFÍA: .....	VIII
	<b>ANEXOS</b>	

## INDICE DE FIGURAS

Figura 1.1: Sistema de Navegación GPS.....	4
Figura 1.2: Funcionamiento de un sistema Dead Reckoning .....	5
Figura 1.3: Localización por triangulación en un sistema por radio navegación .....	6
Figura 1. 4: Partes de un sistema LAV basado en red de datos celular .....	9
Figura 1. 5: Elementos físicos de un sistema LAV basado en red de datos celular.....	10
Figura 1. 6: Esquema de una solución LAV vía radio VHF o UHF.....	11
Figura 1. 7: Hardware necesario para trabajar con fleetSync de Kenwood .....	12
Figura 1. 8: Esquema de un sistema satelital de localización vehicular .....	12
Figura 1. 9: Representación raster de información geográfica .....	16
Figura 1. 10: Representación Vectorial de la información geográfica.....	17
Figura 1. 11: Modelos de representación vector y raster por capas .....	17
Figura 1. 12: Interfaz de usuario Google Earth.....	19
Figura 1. 13: Sistema Básico de Procesamiento Digital de Señal .....	22
Figura 1. 14: Arquitectura Von Neumann.....	24
Figura 1. 15: Arquitectura Harvard.....	24
Figura 1. 16: <i>FDATOO</i> L de <i>MatLab</i> .....	30
Figura 1. 17: Especificaciones para un filtro pasa banda normal .....	30
Figura 1. 18: Especificaciones para un filtro pasa banda más elaborado.....	31
Figura 1. 19: Respuesta en frecuencia de un filtro pasa banda.....	31
Figura 1. 20: Ventana <i>FDATOO</i> L para el diseño del filtro de ejemplo .....	33
Figura 1. 21: Opción Exportar Coeficientes de <i>FDATOO</i> L.....	34
Figura 1. 22: Buffer circular de N localidades de memoria .....	36
Figura 1. 23: Forma de onda almacenada en un buffer circular .....	37
Figura 1. 24: Generación de formas de onda usando un DSP .....	37
Figura 1. 25: Tarjeta de desarrollo Ez Kit Lite ADSP 2181 .....	41
Figura 1. 26: Arquitectura del ADSP 2181 .....	43
Figura 1. 27: Entorno de trabajo del Visual DSP++ 3.5.....	50
Figura 1. 28: Menús importantes del Visual DSP++ 3.5.....	51
Figura 1. 29: Instalación de licencias en <i>VisualDSP++ 3.5</i> .....	52
Figura 1. 30: Licencias instaladas en <i>VisualDSP++ 3.5</i> .....	53
Figura 1. 31: Creación de una nueva sesión en <i>VisualDSP++ 3.5</i> .....	53
Figura 1. 32: Cuadro de diálogo nuevo proyecto.....	54

Figura 1. 33: Opciones de configuración de nuevo proyecto.....	55
Figura 1. 34: Cuadro de diálogo guardar proyecto .....	56
Figura 1. 35: Archivos fuente del proyecto .....	56
Figura 1. 36: Entorno del programa Host del EZ Kit.....	64
Figura 1. 37: Sistema VHF orientado a voz.....	66
Figura 1. 38: Radio base Kenwood.....	67
Figura 1. 39: Radio Portátil Kenwood.....	68
Figura 1. 40: Repetidor Kenwood.....	68
Figura 2. 1: Esquema básico estructural y consideraciones en sistema de transmisión.....	70
Figura 2. 2: Trama de posicionamiento a transmitir en el sistema LAV .....	75
Figura 2. 3: Espectro FM de banda angosta .....	77
Figura 2. 4: Espectro en recepción en respuesta a una señal de ruido blanco.....	78
Figura 2. 5: Entradas de señal de la radio base TK – 7100.....	79
Figura 2. 6: Esquema de Acoplamiento físico en transmisor.....	80
Figura 2. 7: Conformación del Transmisor .....	81
Figura 2. 8: Modulación ASK .....	86
Figura 2. 9: Espectro ASK .....	87
Figura 2. 10: Capacidad ASK.....	88
Figura 2. 11: Modulación FSK.....	88
Figura 2. 12: Espectro FSK.....	89
Figura 2. 13: Capacidad FSK.....	90
Figura 2. 14: Modulación BPSK.....	91
Figura 2. 15: Espectro BPSK.....	92
Figura 2. 16: Capacidad PSK.....	92
Figura 2. 17: Esquema básico estructural y consideraciones en recepción.....	96
Figura 2. 18: Panel posterior de una radio base Kenwood.....	97
Figura 2. 19: Esquema de Acoplamiento físico en Receptor .....	98
Figura 2. 20: Conformación del Receptor.....	99
Figura 2. 21: Carácter asíncrono .....	103
Figura 3. 1: Esquema básico de procesos firmware transmisor .....	104
Figura 3. 2: Esquema básico de procesos firmware receptor.....	105

Figura 3. 3: Diagrama de flujo general de transmisión .....	106
Figura 3. 4: Diagrama de flujo almacenamiento carácter GPS en principal.....	108
Figura 3. 5: Diagrama de flujo almacenamiento GPS en muestreo .....	109
Figura 3. 6: Diagrama de flujo detección y almacenamiento trama NMEA GPRMC.....	112
Figura 3. 7: Diagrama de flujo generación de caracteres de control en transmisión.....	115
Figura 3. 8: Trama GPRMC semi – editada .....	117
Figura 3. 9: Diagrama de flujo módulo de evaluación de apertura automática de canal.....	118
Figura 3. 10: Diagrama de flujo modulación ASK.....	123
Figura 3. 11: Diagrama de flujo inserción bits de inicio y parada.....	124
Figura 3. 12: Proceso de modulación.....	126
Figura 3. 13: Diagrama de flujo inserción caracteres final trama.....	128
Figura 3. 14: Diagrama de flujo modulación FSK.....	130
Figura 3. 15: Diagrama de flujo modulación BPSK .....	132
Figura 3. 16: Diagrama de flujo general de recepción .....	133
Figura 3. 17: Diagrama de flujo filtro digital FIR tiempo Real.....	135
Figura 3. 18: Diagrama de flujo demodulador ASK .....	139
Figura 3. 19: Diagrama de flujo demodulación FSK .....	142
Figura 3. 20: Diagrama de flujo demodulación PSK.....	144
Figura 3. 21: Diagrama de flujo discriminación bits PSK.....	147
Figura 3. 22: Proceso de discriminación de bits .....	148
Figura 3. 23: Diagrama de flujo de validación de información recibida.....	151
Figura 3. 24: Diagrama de flujo de transmisión serial a 9600 bps .....	153
Figura 3. 25: Chequeo de banderas para transmisión serial .....	155
Figura 3. 26: Memorias UVPRM Utilizadas .....	159
Figura 3. 27: Visualización de resultados en Google Earth .....	161
Figura 3. 28: Bases de Datos de localización en Access .....	162
Figura 3. 29: Interfaz tarjeta de desarrollo – radio base .....	163
Figura 3. 30: Vista Superior Circuito Interfaz.....	165
Figura 3. 31: Vista lateral Circuito Interfaz .....	165
Figura 3. 32: Dispositivo GPS con interfaz serial .....	166
Figura 3. 33: Control Automático de ganancia .....	168
Figura 3. 34: Vista superior circuito control automático de ganancia.....	170
Figura 3. 35: Vista Lateral circuito control automático de ganancia.....	170

Figura 3. 36: Vista superior prototipo transmisor.....	172
Figura 3. 37: Vista lateral prototipo transmisor.....	172
Figura 3. 38: Ambiente de trabajo prototipo transmisor.....	173
Figura 3. 39: Vista Superior prototipo receptor.....	175
Figura 3. 40: Vista lateral prototipo receptor.....	175
Figura 3. 41: Ambiente de trabajo prototipo receptor.....	176
Figura 3. 42: Repetidor TKR – 750 de Kenwood.....	178
Figura 3. 43: Ambiente funcionamiento del Repetidor.....	178
Figura 3. 44: Soporte y antena VHF.....	179
Figura 4. 1: Versión primera prototipo transmisor.....	181
Figura 4. 2: Versión primera prototipo receptor.....	181
Figura 4. 3: Primera interfaz tarjeta de desarrollo - radio base.....	182
Figura 4. 4: Primera interfaz tarjeta de desarrollo – Receptor GPS.....	182
Figura 4. 5: Versión segunda prototipo transmisor.....	184
Figura 4. 6: Diagrama de bloques de pruebas realizadas.....	186
Figura 4. 7: Esquema de pruebas a los tipos de modulación sin radio.....	188
Figura 4. 8: Esquema de pruebas a los tipos de modulación vía radio en modo simplex....	188
Figura 4. 9: Forma de onda ASK.....	189
Figura 4. 10: Forma de onda ASK que ha atravesado el canal.....	189
Figura 4. 11: Señal cuasi binaria obtenida de ASK.....	190
Figura 4. 12: Forma de onda FSK.....	191
Figura 4. 13: Forma de onda FSK que ha atravesado el canal.....	191
Figura 4. 14: Señal cuasi binaria obtenida de FSK.....	192
Figura 4. 15: Forma de onda PSK.....	193
Figura 4. 16: Forma de onda que ha atravesado el canal.....	193
Figura 4. 17: Señal cuasi binaria obtenida de PSK.....	194
Figura 4. 18: Esquema de prueba a la sección de discriminación y transmisión serial.....	195
Figura 4. 19: Discriminador de bits no inteligente.....	195
Figura 4. 20: Discriminador de bits inteligente.....	197
Figura 4. 21: Recepción de tramas en estado de pánico.....	198
Figura 4. 22: Recepción de tramas en estado de movimiento.....	198
Figura 4. 23: Recepción de tramas en estado de reposo.....	199

Figura 4. 24: Sustitución de trama perdida.....	199
Figura 4. 25: Esquema General de la pruebas de campo .....	202
Figura 4. 26: Ruta seguida en la primera prueba .....	204
Figura 4. 27: Prototipo transmisor y auto en prueba dos.....	205
Figura 4. 28: Resultados Prueba dos.....	207
Figura 4. 29: Ruta seguida en la prueba dos .....	207
Figura 4. 30: Resultados prueba tres .....	209
Figura 4. 31: Ruta de la prueba tercera .....	210
Figura 4. 32: Prototipo transmisor y auto en prueba cuatro.....	211
Figura 4. 33: Entorno de prueba cuatro .....	211
Figura 4. 34: Ruta recorrida prueba cuatro .....	212
Figura 4. 35: Prototipo transmisor y auto en prueba cinco .....	214
Figura 4. 36: Entorno de prueba cinco .....	214
Figura 4. 37: Límites del área de cobertura.....	215
Figura 4. 38: Mapa de cobertura del sistema LAV .....	217

## INDICE DE TABLAS

Tabla 1.1: Características complementarias de sistemas GPS y Dead Reckoning .....	7
Tabla 1. 2: Clasificación de los DSPs .....	26
Tabla 1. 3: Interrupciones en el ADSP 2181 .....	45
Tabla 2. 1: Tramas GPS Recibidas.....	73
Tabla 3. 1: Comparación de esquemas de modulación.....	157
Tabla 4. 1: Prueba circuito control automático de ganancia .....	201

# **CAPITULO 1**

## **CONCEPTOS GENERALES**

Este capítulo hace una breve descripción de los diferentes sistemas actuales de localización automática vehicular, haciendo un ligero enfoque a los recursos y herramientas que utiliza para lograr su objetivo, permitiendo de esta manera ubicar al lector en el contexto de esta aplicación. Se hace también una breve introducción al procesamiento digital de señal y a cómo opera su sistema básico, estableciéndolo así como principal instrumento del presente proyecto de titulación, por ello se realiza además una breve introducción a la herramienta disponible a utilizar, la tarjeta de desarrollo Ez Kit Lite ADSP – 2181, la cual constituye entonces una parte fundamental del sistema de localización vehicular básico a implementar.

### **1.1. LOCALIZACIÓN AUTOMÁTICA VEHICULAR (LAV)**

#### **1.1.1. DESCRIPCIÓN GENERAL**

Un sistema de localización automática vehicular, en adelante denominado por su acrónimo LAV, como su nombre lo indica permite obtener la ubicación exacta del móvil en determinado instante de tiempo, esto refiriéndose estrictamente al concepto de su nombre, pero un sistema LAV es más que un medio de rastreo, es una herramienta muy útil de información que permite realizar un control logístico adecuado de los móviles que la poseen, como supervisión de sus rutas, inspección del cumplimiento de horarios y manejo general del automóvil, tal es el caso de flotas de vehículos destinadas a una actividad comercial en particular. Inclusive los altos índices delincuenciales respecto al robo de vehículos hacen necesario la implementación de un sistema de rastreo y localización, no solo para aquellos móviles dedicados a actividades de carácter comercial sino también a particulares, lo que permitiría, con un sistema complementario organizado de seguridad, la recuperación del vehículo.

Por lo anterior, empresas de transporte, aseguradoras, empresas de control logístico e inclusive cada día más usuarios particulares ven la necesidad de una solución que permita tener acceso inmediato a la localización de sus vehículos, así como al registro histórico de las ubicaciones de sus activos móviles durante el día, con lo cual se obtienen beneficios como:

- Ahorrar tiempo y dinero en la utilización de rutas eficientes.
- Verificar cumplimiento de itinerarios.
- Mayor seguridad en la transportación.
- Estar al tanto de que móvil está más cercano a un potencial cliente.
- Estar informado periódicamente sobre condiciones relacionadas con velocidad y dirección de desplazamiento.
- Estar informado sobre algún problema que pudiera ocurrir con el o los vehículos.
- Toda la información provista puede ser usada como retroalimentación para la mejora de los servicios prestados, etc.

### **1.1.2. SISTEMAS LAV ACTUALES**

La evolución fantástica de las tecnologías de la información, el auge del mercado de servicios y su mutua complementación, ha propiciado un avance espectacular en cuanto a la utilidad y aplicabilidad de ciertas tecnologías de la ingeniería de las comunicaciones, que si bien ya existían con anterioridad no encontraban una cabida rentable en mercados de aquel entonces, al menos no de forma integral, debida en parte a altos costos y a una utilidad no tan clara entre los potenciales clientes que justificara tal inversión.

La integración de estas tecnologías preexistentes, Sistemas de Posicionamiento Global (GPS), comunicaciones inalámbricas y cartografía digital, se da precisamente por las crecientes, y cada vez más claras necesidades de las empresas prestadoras de servicios de tener un mayor control de sus activos móviles, que redundara en eficiencia en la prestación y por tanto obtener así una ventaja comercial diferenciadora.

Los sistemas LAV actuales tratan de dar solución a estas necesidades, de manera general o dedicada, y su progreso ha ido a la par de la evolución o aparecimiento de nuevas redes que los soportan, de esta manera se hallan diferentes tipos de sistemas LAV.

#### **1.1.2.1. Tipos de Sistemas LAV**

El existir diferentes tipos de sistemas LAV atiende al hecho de la existencia de diversos recursos disponibles para su implementación, a las características y rendimiento exigido de los mismos, así como de los requerimientos por parte del cliente, se tiene así la siguiente clasificación:

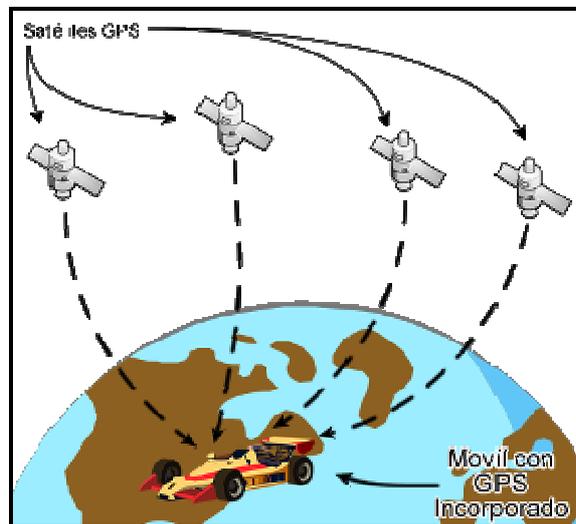
##### ***1.1.2.1.1. Según el origen de los datos de posicionamiento***

Los datos de posicionamiento constituyen la clave para el funcionamiento del sistema LAV, la presente clasificación tiene lugar dado los diferentes sistemas de navegación que pueden ser utilizados como origen de estos datos.

##### ***1.1.2.1.1.1. Sistemas basados en GPS***

La mayoría de los sistemas LAV tienen como fuente de datos de posicionamiento los originados en un sistema GNSS (*Sistema Global de Navegación por Satélite*), el más conocido, ampliamente utilizado y completamente operativo es el GPS (*Sistema de posicionamiento Global*), que utiliza una red de veinte y cuatro satélites agrupados en

seis órbitas, el dispositivo GPS utiliza tres satélites para calcular su posición mediante triangulación y uno determina la altitud, es decir utiliza una de seis órbitas.



**Figura 1.1:** Sistema de Navegación GPS

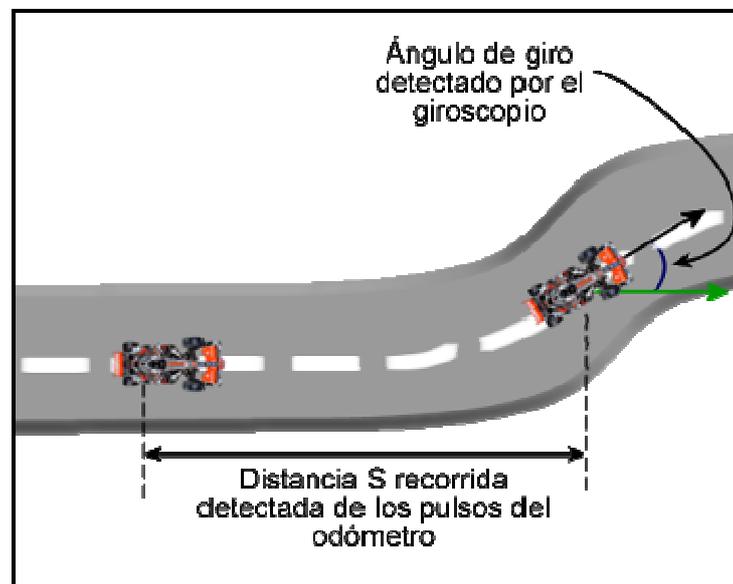
Los dispositivos GPS no solamente entregan información relacionada a su ubicación (Latitud y Longitud), entre los básicos también se encuentran datos relacionados con su velocidad, dirección de movimiento, hora y fecha actualizada, es decir información completa para la ayuda en navegación.

#### **1.1.2.1.1.2. Sistemas no basados en GPS**

Estos sistemas no tienen como origen de datos de navegación los obtenidos mediante la red de satélites GPS, basan su funcionamiento en la información proporcionada por otros métodos conocidos como, *Dead reckoning (DR)* y Radio navegación, a continuación se describe brevemente cada una de las tecnologías mencionadas:

- **Dead Reckoning:** Es el proceso de estimación de la posición actual sobre la base de una posición previamente determinada, tomando en consideración la rapidez, dirección del movimiento, así como el tiempo transcurrido, la posición

actual es entonces calculada en base a sencillas fórmulas trigonométricas, en la implementación de un sistema LAV considerando esta tecnología, cada uno de los autos incorpora una brújula digital y sensores en las ruedas, la tasa de error promedio en la estimación de la posición actual suele ser del uno o dos por ciento, esto provoca una acumulación de los errores ya que el cálculo de la posición vigente se calcula en base a la inmediatamente anterior; un ejemplo de un sistema LAV que implementa este método de navegación es el *CROSSCHECK XR* de la empresa *DATUMCOM*<sup>2</sup>.

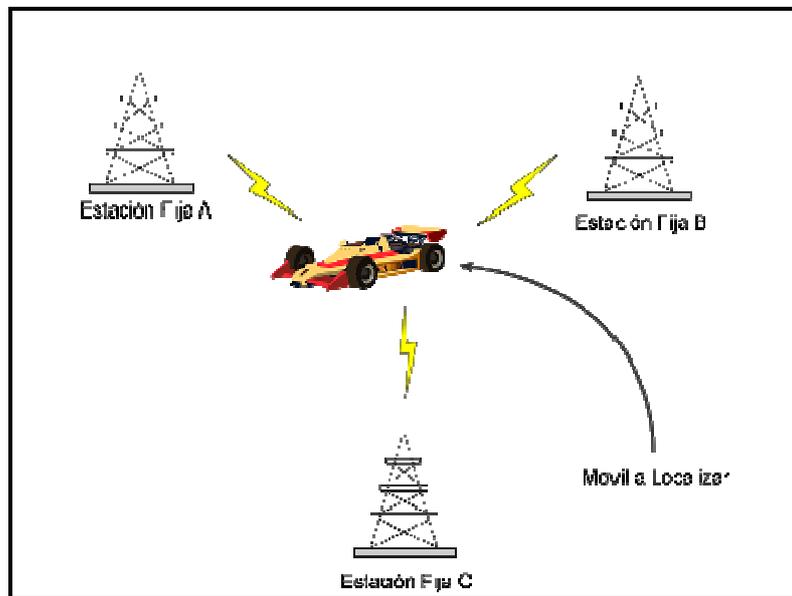


**Figura 1.2:** Funcionamiento de un sistema Dead Reckoning

- **Radio Navegación:** Un receptor de radio navegación incorporado en un vehículo hace mediciones de tiempo en la propagación de señales, denominadas *beacon*, provenientes de al menos tres ubicaciones fijas cuyas posiciones exactas son conocidas, estas medidas determinan cuán lejos está el dispositivo receptor de dichas estaciones fijas, con esta información se realiza el cálculo aproximado de la posición del vehículo mediante triangulación, un

<sup>2</sup> <http://www.datumcom.com/np.cfm#axr>

ejemplo de estos sistemas de navegación es el denominado LORAN<sup>3</sup> (*Long Range Navigation*).



**Figura 1.3:** Localización por triangulación en un sistema por radio navegación

#### 1.1.2.1.1.3. *Combinación de sistemas GPS y no GPS*

El objetivo de la combinación de sistemas GPS y no GPS es el de perfeccionar la exactitud de los datos de posicionamiento, así como de prolongar su disponibilidad en ausencia de uno de ellos, típicamente del sistema satelital. De esta manera se han combinado sistemas de navegación GPS con sistemas de tipo *Dead Reckoning*, con el objetivo de minimizar los efectos de los cortes de señal del GPS, en sitios cerrados como túneles o parqueaderos, o en días en que las condiciones climatológicas no permiten el enlace satelital necesario para la operación del receptor de posicionamiento global, así como también compensar la no precisión, debido a la acumulación de errores, en el cálculo de la posición del sistema *Dead Reckoning*.

La siguiente tabla muestra las características complementarias de estos dos sistemas:

<sup>3</sup> <http://www.loran.org/>

<b>Debilidades del sistema GPS</b>	<b>Fortalezas del sistema Dead Reckoning</b>
<i>Dependencia de condiciones climatológicas</i>	<i>No dependiente de condiciones climatológicas</i>
<i>Puede haber errores debido al efecto "multipath"</i>	<i>Es inmune a efectos "multipath"</i>
<b>Fortalezas del sistema GPS</b>	<b>Debilidades del sistema Dead Reckoning</b>
<i>Posicionamiento Absoluto</i>	<i>Necesita de una posición absoluta como punto de partida</i>
<i>Errores no se acumulan</i>	<i>Los errores se acumulan</i>

**Tabla 1.1:** Características complementarias de sistemas GPS y Dead Reckoning

Un ejemplo comercial de este tipo de sistemas que integran estos dos métodos de navegación es el producto *NTSTracer*<sup>4</sup> de la empresa NTS.

#### ***1.1.2.1.2. Según la característica de tiempo real***

Los sistemas LAV también se los puede clasificar tomando en cuenta si transmiten o no la información de la posición el preciso instante en que se genera, la aplicabilidad de cada uno de estos sistemas depende de los objetivos perseguidos por el cliente que los contrata y los considera adecuados.

##### ***1.1.2.1.2.1. Sistemas en línea***

Los sistemas LAV en línea son diseñados para comunicar el dato de posición de manera instantánea, para lo cual necesitan apoyarse en un sistema de comunicaciones inalámbricas que tenga algún soporte mínimo para el transporte de datos, de esta manera se encuentran en el mercado sistemas LAV que utilizan redes celulares, de radiofrecuencia y satelitales.

Los sistemas en línea son útiles en aquellos casos en los cuales es muy necesario conocer la posición vehicular en un instante dado, posibilitando la toma de decisiones en tiempo real según sea necesario, un ejemplo práctico es el de una compañía organizada de taxis la cual necesita obligadamente conocer la posición exacta de

<sup>4</sup> [http://www.nts-world.com/products\\_tracer.htm](http://www.nts-world.com/products_tracer.htm)

alguno de sus miembros para designar al más próximo alguna tarea de transporte al lugar desde donde se requiere el servicio.

#### ***1.1.2.1.2.2. Sistemas fuera de línea***

Quienes hacen uso de este sistema no tienen como objetivo el conocer de manera inmediata la posición de sus activos móviles, sino conocer al final de cierto período, el registro histórico de sus ubicaciones con fines de control que no implican decisiones en tiempo real.

Los dispositivos vehiculares ya no tienen la necesidad de formar parte de un sistema de comunicaciones, sino que incorporan un módulo de memoria que les permite almacenar las diferentes posiciones ocupadas por el móvil en instantes de tiempo previamente establecidos y programados.

Ejemplo de un sistema LAV fuera de línea es el ofrecido por la empresa *TELESPIAL SYSTEMS INC.* Denominado *trackstick*<sup>5</sup>.

#### ***1.1.2.1.3. Según la red que los soporta***

Como se había mencionado en los sistemas LAV en línea, estos necesitan apoyarse necesariamente en sistemas de comunicaciones inalámbricas que permitan su implementación en tiempo real. Cabe destacar que el éxito de un sistema LAV es, por supuesto, la gran cobertura que debe tener, así el área cubierta es tan amplia como la red que los soporta, de ahí la importancia de esta clasificación.

##### ***1.1.2.1.3.1. Sistemas sobre red de datos celular***

En principio la red de telefonía celular fue concebida con el objetivo de transmitir eficientemente información relacionada solamente con voz, hace pocos años se ve la necesidad de transmitir y recibir datos, impulsado esto por la idea de la convergencia tecnológica, para lo cual era necesario realizar una re ingeniería de la plataforma que la soporte, es así que hoy en día se hallan sistemas modernos celulares de datos

---

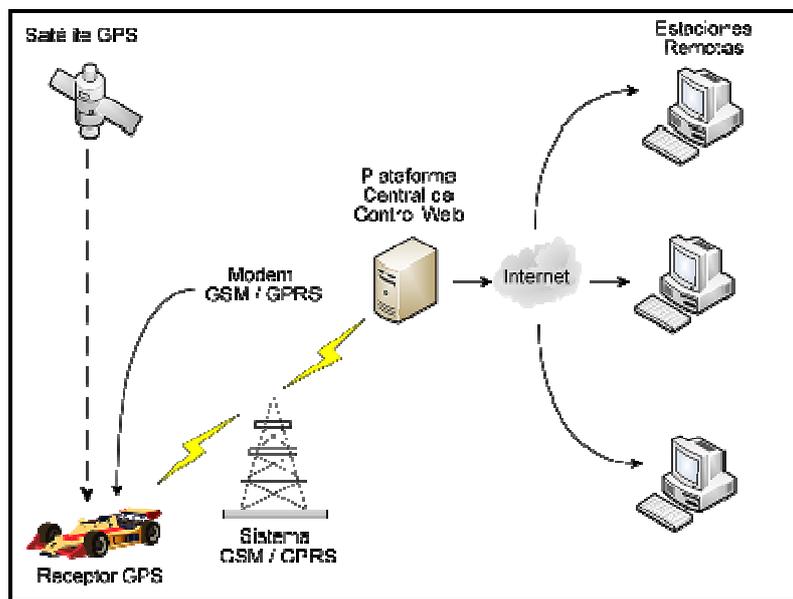
<sup>5</sup> <http://www.trackstick.com/>

como *GPRS*, *EDGE*, *CDMA/1xRTT*, denominados actualmente como Banda ancha móvil.

Esta plataforma diseñada para datos es propicia para la implementación de un sistema LAV, dada no solamente las condiciones adecuadas para la transmisión de información de navegación sino también por la ventaja de cobertura, que incluye normalmente hasta tramos rurales como carreteras, lo cual es importantísimo para un sistema LAV exigente.

Estas características han hecho de las redes de datos celulares las más populares para dar soporte a sistemas LAV, sin que se haya abandonado por completo otras redes que también resultan apropiadas, de las cuales se hablará posteriormente.

Estos sistemas hacen uso de modernos módems celulares, los cuales les permite interactuar con la red de datos celular, a continuación se muestra un ejemplo típico de las partes constitutivas de uno de estos sistemas:



**Figura 1. 4:** Partes de un sistema LAV basado en red de datos celular

Como muestra la figura 1.4 un sistema LAV que utiliza la red celular de datos hace uso de una interconexión a internet facilitando la visualización de la información vehicular no solamente desde una central única, sino que se puede acceder a la

misma desde cualquier lugar remoto que posea una PC con conexión a la gran red mundial, ya que la información es depositada por el sistema GSM/GPRS en un servidor web.



**Figura 1. 5:** Elementos físicos de un sistema LAV basado en red de datos celular

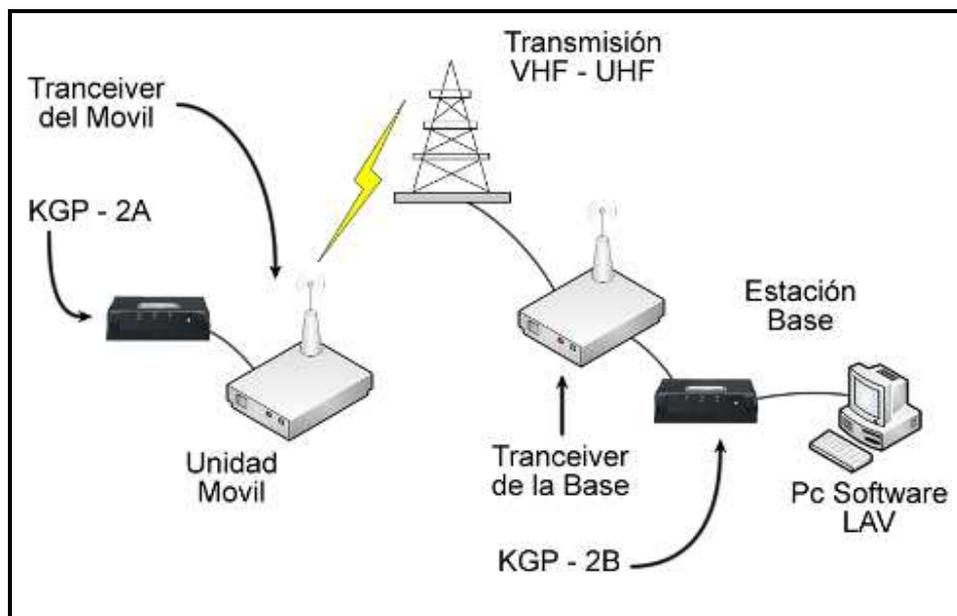
La figura 1.5 muestra el sistema LAV llamado *StarFinder AVL*<sup>6</sup> donde se destaca principalmente el módem GSM/GPRS, con todas sus interfaces, así como el dispositivo receptor GPS.

#### ***1.1.2.1.3.2. Sistemas vía radiofrecuencia***

Cuando la aplicación, en la que se usen móviles a monitorear, se desarrolla dentro del perímetro urbano, utilizar enlaces de radio VHF o UHF se convierte en una opción apropiada, por cuanto resulta una opción menos costosa que su contraparte celular.

En el mercado existen soluciones que pueden adaptarse a los sistemas de radio existentes, tal es el caso de Kenwood, que ofrece los módulos denominados *KGP - 2A/2B*<sup>7</sup>, el esquema de esta solución es el siguiente:

<sup>6</sup> [http://www.laipac.com/gps\\_starfinder1\\_esp.htm](http://www.laipac.com/gps_starfinder1_esp.htm)



**Figura 1. 6:** Esquema de una solución LAV vía radio VHF o UHF

El sistema está basado en los datos de navegación provistos por el dispositivo GPS, los módulos *KGP – 2A* y *KGP – 2B* implementan los módems que se encargan de adecuar la información a enviar a la estación base para que puedan atravesar el canal inalámbrico de comunicaciones.

Existe otra solución de Kenwood para poder implementar un sistema LAV, se trata de lo que denominan *FleetSync*<sup>8</sup>, que es un sistema de señalización digital que permite establecer una plataforma sobre la cual manejar el envío y recepción de datos, a diferencia del sistema LAV descrito por la figura 1.6 que se monta sobre uno existente y que maneja solamente información relacionada con Voz.

El uso de esta plataforma para el manejo de datos implica también usar nuevo hardware como el que se muestra a continuación:

<sup>7</sup> [http://www.kenwoodusa.com/Communications/Land\\_Mobile\\_Radio/Fleet\\_Management/KGP-2A-2B](http://www.kenwoodusa.com/Communications/Land_Mobile_Radio/Fleet_Management/KGP-2A-2B)

<sup>8</sup> [http://www.kenwoodusa.com/Communications/Land\\_Mobile\\_Radio/Fleet\\_Management/KDS-100](http://www.kenwoodusa.com/Communications/Land_Mobile_Radio/Fleet_Management/KDS-100)

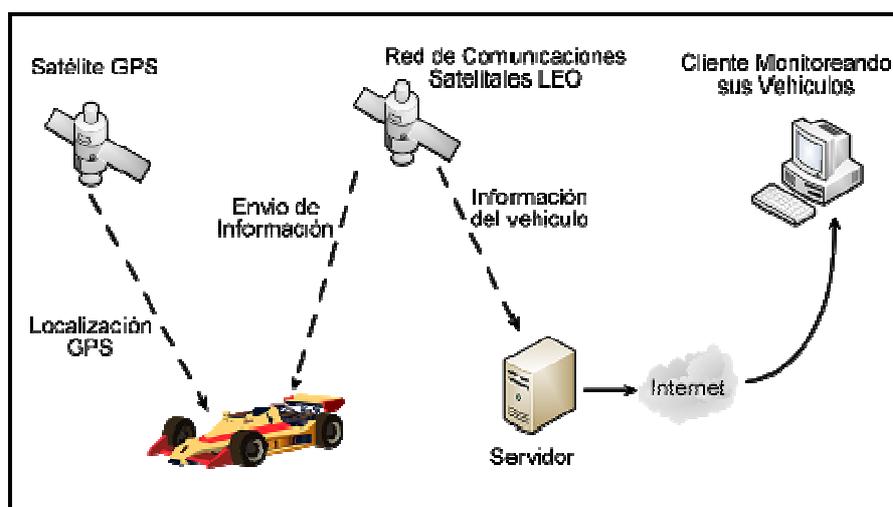


**Figura 1. 7:** Hardware necesario para trabajar con fleetSync de Kenwood

Las limitaciones de un sistema LAV vía radio frecuencia viene dada por el área de cobertura que se puede alcanzar, dependiente de la potencia emitida por las antenas apropiadas y por condiciones geográficas donde se desarrolla la aplicación.

#### *1.1.2.1.3.3. Sistemas satelitales*

Para sistemas LAV mucho más exigentes en cuanto a cobertura, donde la red celular no es suficiente, se tienen en el mercado opciones que hacen uso de redes satelitales de baja altura, típicamente conocidas como redes LEO (*Low Earth orbit*), tales sistemas deben incluir en el hardware necesario módems satelitales que adecuan la señal para lo que se conoce como enlace de subida y, en la contraparte receptora, el enlace de bajada. A continuación se presenta un esquema de su funcionamiento:



**Figura 1. 8:** Esquema de un sistema satelital de localización vehicular

El esquema de la figura 1.8 es similar al mostrado para el sistema LAV que usa una red celular, variando solamente la forma de comunicar la información relacionada con el vehículo al servidor Web.

La principal desventaja de este modelo LAV es, al igual que para el dispositivo GPS, las condiciones climatológicas favorables que requiere, para la no errónea transmisión y recepción de la información.

## **1.2. SISTEMAS DE INFORMACIÓN GEOGRÁFICA (SIG)**

Para hacer inteligible y más aprovechable la información de navegación transmitida o almacenada por los correspondientes dispositivos, la localización automática vehicular hace uso de un recurso fundamental, los sistemas de información geográfica, de tal manera que los datos de localización no resulten tan abstractos y se relacionen unívocamente con la realidad a través de un mapa digitalizado, facilitando aspectos de planificación y gestión, de ahí la importancia de analizar brevemente a los sistemas de información geográfica.

Un sistema de información geográfica (*SIG* o *GIS* por su acrónimo en inglés), es cualquier sistema de información que provea la capacidad de integrar, almacenar, editar, analizar, compartir y mostrar la información geográficamente referenciada con el propósito de ayudar a resolver problemas de planificación y gestión.

### **1.2.1. IMPORTANCIA ACTUAL**

La necesidad de información de tipo geográfica ha sido relevante a lo largo de la historia de la humanidad, esta afirmación es indiscutible y pueden darse entonces muchos ejemplos de ello, desde la época del hombre de las cavernas hasta la presente fecha, la tendencia siempre ha sido asociar una imagen territorial con un aspecto de información, un ejemplo emblemático es proporcionado por la historia cuando en 1854 el Dr. John Snow cartografió la incidencia de casos de cólera en un distrito de Londres, lo cual le permitió hallar rápidamente la causa del mal, un pozo de agua contaminado.

Desde ese entonces hasta hoy han acontecido muchos sucesos que han hecho que los sistemas SIG evolucionen y se perfilen como una herramienta indispensable en innumerables disciplinas, fundamentalmente estos sucesos están ligados al surgimiento y auge de la ciencia informática.

Así entonces la información visual provista por un SIG puede ser utilizada en investigaciones científicas, gestión de recursos, gestión de activos, evaluaciones de impactos ambientales, planificación urbana, sociología, geografía histórica, marketing, arqueología, epidemiología, logística, etc., constituyéndose de esta manera en un instrumento imprescindible. Por ejemplo un SIG podría permitir a los grupos de emergencia calcular fácilmente los tiempos de respuesta en caso de un desastre natural así como las rutas adecuadas de evacuación.

### **1.2.2. FUNCIONAMIENTO DE UN SIG**

Un SIG funciona como una base de datos inteligente con información geográfica que se halla asociada a los objetos gráficos. Así si se señala un objeto se podrán conocer sus atributos, e inversamente, preguntando a la base de datos por un objeto se puede saber su localización en la cartografía.

Típicamente la información de interés se encuentra distribuida en diferentes capas temáticas almacenadas de manera independiente de tal forma que el usuario podrá hacer uso de ellas como crea conveniente simplemente activándolas, ejemplos de capas son: vías de comunicación, núcleos de población, límites fronterizos, Relieves, red fluvial, usos de suelo, etc. Todo esto permite realizar un análisis completo cuyo alcance depende de la aplicación a la cual esté sirviendo el Sistema de Información Geográfica.

### **1.2.3. TÉCNICAS UTILIZADAS EN LOS SISTEMAS DE INFORMACIÓN GEOGRÁFICA**

#### **1.2.3.1. Creación y obtención de datos**

Típicamente la información de imagen proviene de satélites u aviones, las cuales se someten a un proceso de digitalización y orto – rectificación, con lo cual se obtiene una orto-fotografía, que es una representación fotográfica de una zona de la superficie terrestre, en que todos los elementos presentan la misma escala, libre de errores y deformaciones causados por la perspectiva desde la cámara, la altura o la velocidad a la que se mueve la misma. Todo esto provee la misma validez de un plano cartográfico tradicional con lo cual es posible realizar mediciones exactas.

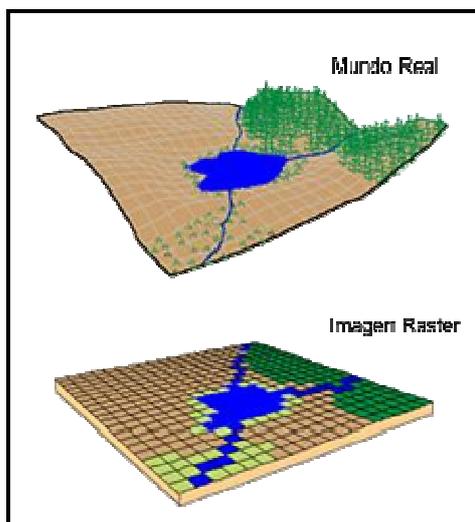
#### **1.2.3.2. Representación de datos**

Los datos digitalizados representan los objetos del mundo real, estos se los puede almacenar y representar de dos formas, conocidas como raster y vectorial.

##### ***1.2.3.2.1. Raster***

Una imagen raster, es aquella cuya unidad de representación es lo que se conoce como píxel o celda, es decir una combinación de estos elementos conformará una imagen completa.

Los datos raster se componen de filas y columnas de píxeles, asignando a cada uno de estos, valores únicos de información. Según la temática de la capa del SIG, estos valores corresponderán a diferentes aspectos geográficos relevantes como la conformación del mapa base, en este caso el valor asignado es un color, o información discreta relativa a las condiciones del entorno geográfico, entonces dicho valor puede ser referente a información de suelos, temperatura promedio, altura sobre el nivel del mar, posición exacta, etc. En un modelo raster cuanto mayor sea la dimensión de la celda menor es la precisión o detalle de la representación del espacio geográfico y de sus características.



**Figura 1. 9:** Representación raster de información geográfica

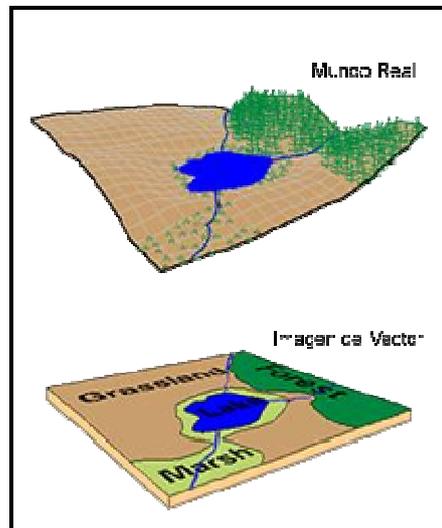
#### **1.2.3.2.2. Vectorial**

A diferencia del modelo *raster*, el vectorial ofrece una mayor precisión en la representación de la realidad, tratando de mantener al máximo posible las características geométricas de las figuras, cada una de estas figuras está vinculada a una base de datos que describe sus atributos, relativos a altitud, temperatura, posición, tipo de suelo, etc. según sea la aplicación en curso.

Para modelar digitalmente las entidades del mundo real se utilizan tres elementos geométricos:

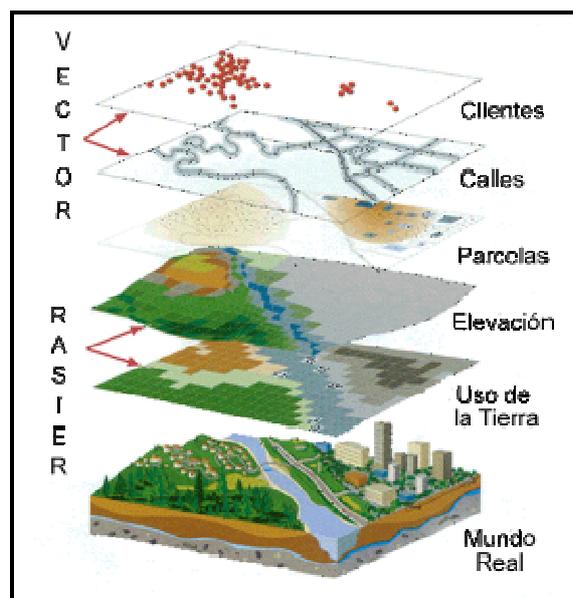
- *El punto:* se utilizan para representar zonas a una escala pequeña, por ejemplo ciudades en un mapa del mundo, también en la ubicación de puntos de interés, como pozos, cumbres de elevaciones, etc.
- *La línea:* Se usa para representar rasgos lineales, por ejemplo ríos, caminos, curvas de nivel, etc. o estructuras geográficas que en escala pequeña pueden ser representados linealmente. De esta forma ya se puede realizar mediciones relativas a distancia.
- *Polígonos:* Sirven para representar entidades geográficas que en una escala adecuada representan un área de la superficie de la tierra, por ejemplo lagos,

edificios, provincias; son las estructuras geométricas que mayor información aportan y en ellos se puede medir perímetros y áreas.



**Figura 1. 10:** Representación Vectorial de la información geográfica

Evidentemente la decisión de utilizar uno u otro tipo de representación dependerá de la aplicación, lógicamente en un sistema de localización vehicular será deseable tener una precisión óptima la cual es provista solamente por la representación vectorial.



**Figura 1. 11:** Modelos de representación vector y raster por capas

#### 1.2.4. SOFTWARE SIG

En la actualidad existen varios proveedores de completas soluciones SIG, algunos ejemplos son *Intergraph*<sup>9</sup>, *Google*<sup>10</sup>, *Autodesk*<sup>11</sup>, etc. la mayoría de estos ofrecen soluciones de pago, es decir hay que adquirir derechos de utilización de su software además de tener cierto grado de conocimiento en su utilización, sin embargo la tendencia actual es que estas herramientas sean hasta cierto punto gratuitas para acceder a la información geográfica y de intuitiva manipulación, de esta manera el acceso del público en general a los geodatos está dominado por la disposición de recursos en línea, como el *Google Earth* cuya versión gratuita ha revolucionado la utilización de software SIG calificándolo incluso de una herramienta de “Mapeo para masas”.

##### 1.2.4.1. Google Earth

*Google Earth* es un software SIG de tendencia actual que muestra imágenes satelitales variando la resolución de la superficie terrestre, permitiendo visualizar en detalle países, provincias, ciudades, pueblos, accidentes geográficos, etc. desde una vista perpendicular superior y a escala, ya que para su elaboración ha usado técnicas orto – fotográficas y de representación vectorial.

Permitiendo de esta manera obtener todas las ventajas de un software SIG de propósito general, proporcionando en este sentido información geográfica relacionada con:

- Características de sitios específicos como ubicación geográfica exacta, altitud sobre el nivel del mar.
- Además de destacar sitios de interés turístico, nombres de carreteras principales, secundarias, caminos vecinales, etc.
- Búsqueda de sitios en la base de datos geográficos, entre otras características.

---

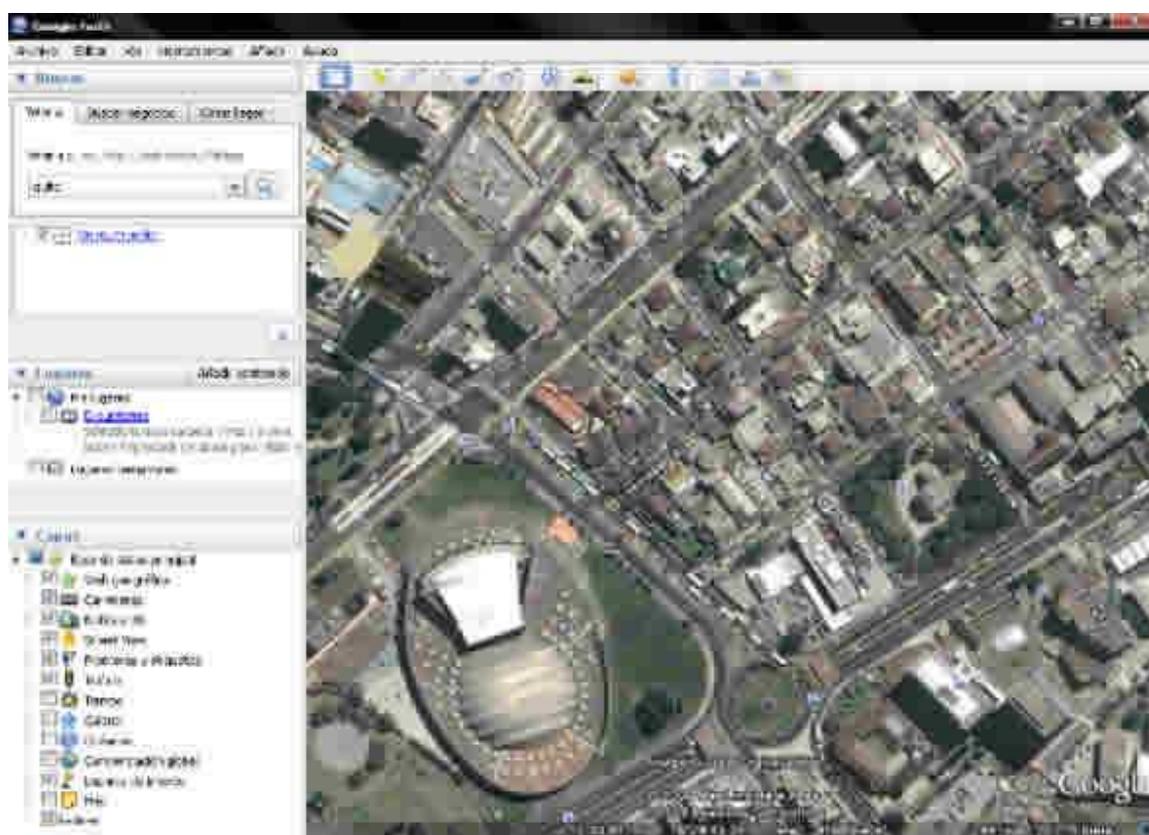
<sup>9</sup> <http://www.intergraph.com/learnmore/sgi/government/gis-software.aspx>

<sup>10</sup> <http://www.google.es/earth/index.html>

<sup>11</sup> <http://usa.autodesk.com/adsk/servlet/item?siteID=455755&id=6908788>

Además de herramientas relacionadas con importación de datos GPS, medición de distancias y un interesante simulador de vuelo.

Actualmente este software es muy utilizado en diversas áreas, por ejemplo los medios de comunicación para ubicar lugares exactos donde se genera información, en arquitectura e ingeniería para realizar estudios de viabilidad técnica, localización de obras, análisis de impacto ambiental, desarrollo urbano, etc., de manera general es una herramienta fundamental en la planificación y gestión.



**Figura 1. 12:** Interfaz de usuario Google Earth

La precisión de calles y avenidas mostrada hace de este software el más idóneo para especificar la posición en tiempo real de los vehículos y su cercanía a sitios de interés, estableciendo rutas y móviles apropiados para cumplir con determinada acción en un tiempo mínimo planificado.

Son estas razones además de su gratuidad y fácil manipulación las que lo convierten en el software SIG apropiado para hacer uso del mismo en el presente proyecto de titulación.

### **1.3. PROCESAMIENTO DIGITAL DE SEÑAL**

El procesamiento digital es un área importante de la ingeniería que ha permitido revolucionarios avances en cualquier campo de la ciencia que incluya el tratamiento de señales provenientes del entorno, abarcando de esta manera un sinnúmero de disciplinas que se han visto beneficiadas directamente, por ejemplo las telecomunicaciones, el control, la exploración del espacio, la medicina y geología solo por nombrar unas pocas.

El procesamiento de señales trata de la representación, transformación y manipulación de eventos físicos de notable interés, cuando se refiere al procesado digital, se especifica la representación mediante secuencias de números de precisión finita y el procesado respectivo se lo realiza utilizando un computador digital.

A menudo es deseable que estos sistemas de procesamiento digital de señal funcionen en tiempo real, lo que significa que el sistema en tiempo discreto se implementa de tal suerte que las muestras de salida se calculen a la misma velocidad a la que se muestrea la señal de entrada en tiempo continuo, son muchas las aplicaciones que requieren de esta especificación. El tratamiento en tiempo discreto y en tiempo real de señales en tiempo continuo es ya una práctica común en sistemas de control, de comunicaciones, radar, sonar, codificación, imágenes, voz, biomedicina, sismología y un largo etcétera.

El procesamiento digital de señales ha existido desde hace algunas décadas, pero la existencia de computadores de capacidad limitada y de alto costo restringían severamente las posibles aplicaciones en tiempo real y no real, sin embargo la invención de algoritmos de procesamiento digital eficientes, como la transformada rápida de Fourier, y el avance vertiginoso en la industria de los semiconductores,

representado por la invención de potentes microcontroladores, posibilita hoy en día realizar cualquier aplicación de procesamiento digital de señal en tiempo real.

De esta manera las ventajas en el procesado digital de señal son las siguientes:

- **Repetitividad:** Un sistema digital es inherentemente repetitivo, por ejemplo si cien ordenadores realizan una misma secuencia de sumas todos ellos arrojarán un idéntico resultado, en contraste, si aplicamos una señal a cien circuitos analógicos, contruidos con elementos de idénticas especificaciones, muy probablemente no se obtengan las mismas respuestas en ninguno de ellos.
- **Elevada estabilidad térmica:** Es evidente que los componentes no discretos de un circuito analógico presentan características muy dependientes de la temperatura, pudiendo cambiar drásticamente su respuesta afectando sensiblemente el rendimiento para el cual fue diseñado dicho circuito, esto deriva en una ventaja implícita de los sistemas digitales, *Precisión*.
- **Reprogramabilidad:** Es este un indicativo de flexibilidad, pues se puede realizar configuraciones de hardware que pueda ser programado para realizar una amplia variedad de tareas de procesamiento digital de señal, simplemente cambiando el programa que gobierna el funcionamiento.
- **Almacenamiento:** El procesamiento digital de señales ha revolucionado la concepción del almacenamiento de datos, con lo cual en la actualidad es posible acumular información de cualquier tipo en gigantescas cantidades.

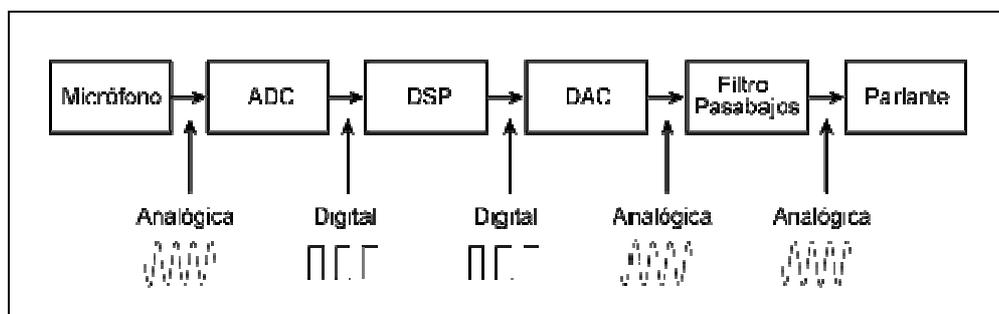
Son estas algunas de las ventajas del procesamiento digital de señales y que son suficientes para darse cuenta de la relevante importancia que su aprendizaje y aplicación implica.

### 1.3.1. SISTEMAS DE PROCESAMIENTO DIGITAL DE SEÑAL

#### 1.3.1.1. Esquema Básico

Un sistema de procesamiento digital de señal consiste básicamente de tres pasos: conversión de la señal analógica en digital, procesamiento de la señal digital y finalmente, conversión de la señal analizada y procesada a una forma analógica, la

figura mostrada a continuación se refiere a un sistema de procesamiento digital de señales aplicado a un caso particular, el tratamiento de señales en el rango de frecuencias audible:



**Figura 1. 13:** Sistema Básico de Procesamiento Digital de Señal

Cómo funciona el sistema descrito en la figura anterior se explica a continuación:

Dado que la amplitud de la señal analógica de entrada varía con el tiempo, se usa un circuito de muestreo y retención (*Sample and Hold*) que muestrea dicha señal en intervalos periódicos de tiempo y mantiene constante el valor de dicha muestra a la entrada del convertidor analógico – digital (*ADC*), para llevar a cabo la conversión a digital.

La salida del convertidor *ADC* es una cadena de datos binarios que posteriormente son tratados por el procesador digital de señal, que implementa el algoritmo deseado, eventualmente la salida del procesador digital es convertida en una señal analógica por el convertidor digital – analógico (*DAC*), finalmente un filtro pasa bajos elimina los componentes de frecuencia indeseables para que a la salida se tenga una señal completamente analógica.

La relación funcional entre los elementos de un sistema de procesamiento digital de señal (*DSP*) debe ser, por supuesto correspondiente, como se mencionó párrafos arriba, las aplicaciones que hacen uso de sistemas *DSP* típicamente son de tiempo real por tanto deben procesar y generar información al tiempo que la capturan, de esta manera los conversores analógico – digital deben adquirir la información lo suficientemente rápido para capturar todas las fluctuaciones relevantes de la señal, si

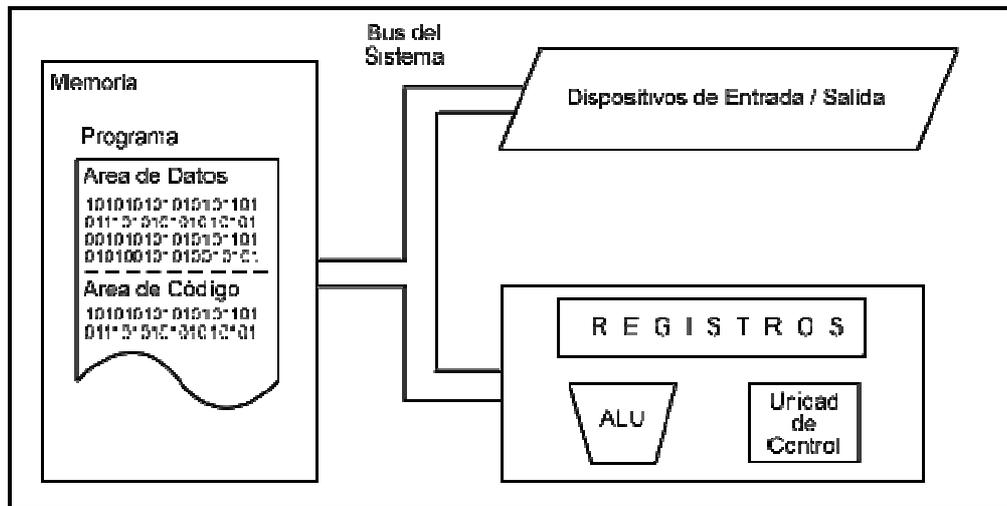
el *ADC* es lento se perderá información; de idéntica forma el microcontrolador *DSP* debe trabajar rápido para no perder información que le llega desde el *ADC* y además cumplir con el respectivo tratamiento de señal, así entonces dependiendo del tipo de señal a procesar se deberá escoger los elementos *ADC – DSP – DAC* adecuados. Por ejemplo, un sistema estéreo maneja sonidos de hasta 20 KHz, por lo tanto el *ADC* debe manejar frecuencias de muestreo de hasta 40 KHz según el teorema de Nyquist, y el *DSP* deberá ser capaz de procesar alrededor de millones de operaciones por segundo. Otras señales, tales como transmisiones por satélite están en el orden de los *Gigahertzios* por lo que requieren un procesamiento a mayor velocidad.

De esta manera todos los elementos de un sistema de procesamiento digital de señal juegan un rol importante, pero el más destacable de todos es el microcontrolador *DSP* ya que este es el que implementa en la práctica el algoritmo de tratamiento de señal.

#### **1.3.1.2. Microcontrolador DSP**

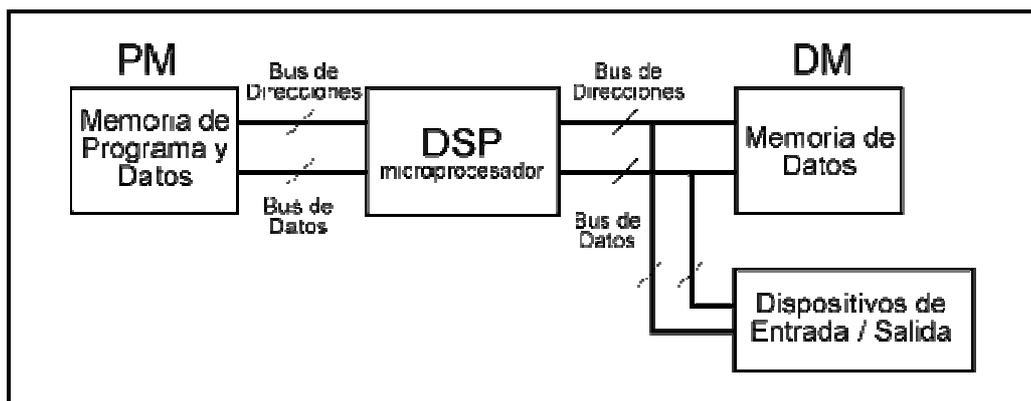
Los microcontroladores *DSP* son dispositivos programables creados con un fin dedicado, realizar operaciones matemáticas típicas del procesamiento digital de señales, para lo cual su configuración interna difiere de los microcontroladores de propósito general como se describirá y explicará a continuación:

Las mencionadas operaciones típicas en un sistema de procesamiento digital de señal son las multiplicaciones y las sumas, para realizar la multiplicación de dos números usando un procesador convencional se requeriría acceder a memoria más de una vez para obtener los datos y la instrucción, ya que estos se encuentran en una misma sección de memoria, a esta forma conjunta de almacenar datos y programa se le denomina arquitectura *Von Neumann* y es la implementada en la mayoría de procesadores comunes. Un esquema simple de lo mencionado se presenta a continuación:



**Figura 1. 14:** Arquitectura Von Neumann

Como se muestra en la figura, dicha arquitectura cuenta con un solo bus para ser ocupado por datos y programa, la limitación es evidente, por cuanto se requieren varios accesos a memoria para llevar la instrucción a la unidad central de proceso, si dicha instrucción maneja datos de memoria, se deben realizar otros accesos para llevar, operar y almacenar los datos, lo cual resulta en un congestionamiento del bus. La solución prevista por los microcontroladores DSP viene impuesta por la llamada arquitectura *Harvard*:



**Figura 1. 15:** Arquitectura Harvard

La unidad central de proceso está conectada a dos memorias por intermedio de dos buses separados, una de las memorias contiene las instrucciones de programa y

eventualmente datos, y la otra solo contiene datos, es por esto que el tiempo de ejecución mejora notablemente, ya que se tiene acceso a instrucciones y datos simultáneamente.

Ambos buses son totalmente independientes por lo que la unidad central de proceso puede estar accediendo a los datos para completar la ejecución de una instrucción y, al mismo tiempo, estar leyendo la próxima instrucción a ejecutar.

Otra de las características importantes y básicas de un microcontrolador DSP es su capacidad de realizar operaciones de multiplicación y acumulación en un solo ciclo de máquina, ya que poseen una unidad computacional implementada en hardware para tales efectos llamada *MAC (Multiplier - Accumulator)*, además de la inclusión de estructuras también implementadas en hardware denominadas *DAGs (Data Address Generators)* que se encargan automáticamente del direccionamiento indirecto.

Todo lo mencionado anteriormente establece una arquitectura que soporta un flujo de información a altas velocidades desde y hacia las unidades computacionales incluyendo los accesos a memoria, lo cual permite al DSP acceder a los datos, procesar la instrucción respectiva que maneja dichos datos y guardar el resultado en memoria, todo esto en un solo ciclo de máquina.

Es así entonces que este dispositivo dedicado puede ejecutar algoritmos de procesamiento digital de señales complejos en tiempo real.

Tal como se plantea la configuración de un microcontrolador DSP podría pensarse que constituye la solución para todas las aplicaciones de procesamiento digital de señal, sin embargo dependiendo de la aplicación puede encontrarse diversos tipos de microcontroladores DSP que varían su precisión expresada a través de su rango dinámico.

El rango dinámico es el intervalo de números, desde el más pequeño al más grande, que el microcontrolador DSP puede procesar en el transcurso de la aplicación, por ejemplo, para representar una forma de onda entera de una señal en particular, el DSP debe ser capaz de poder representar tanto su valor mínimo como su máximo, si

no es capaz de manejar todo el rango dinámico necesario para la aplicación ocurrirán desbordamientos produciendo errores en los cálculos.

En este sentido la capacidad de un microcontrolador está en función de su ancho de datos (*número de bits manipulados*) y el tipo de aritmética que posee (*punto fijo o flotante*). Un procesador de 32 bits tiene un ancho mayor que uno de 24 bits y este a su vez uno mayor que otro de 16 bits. DSPs de punto flotante tienen un rango mayor que los de punto fijo; cada tipo de microcontrolador DSP es ideal para determinados rangos de aplicaciones en particular, DSPs de 16 bits son ideales para sistemas de audio, DSPs de punto flotante de 32 bits son necesarios para procesamiento de imágenes, gráficos en tres dimensiones y simulaciones científicas. De esta manera en la actualidad existen diferentes fabricantes, que manufacturan DSPs de diversas características, la tabla siguiente muestra varios fabricantes y señala algunos de sus recientes productos clasificándolos por el número de bits que manejan.

Fabricante	DSP 16 bits	DSP 24 bits	DSP 32 bits
Analog Devices <sup>12</sup>	ADSP – 2191(21xx) ADSP – 21991 (21xx) ADSP-BF512 (Blackfin)	-	ADSP – 21261 (SHARC) ADSP – 21362 (SHARC) ADSP-21160N (SHARC)
Texas Instruments <sup>13</sup>	TMS320C1x TMS320C2xx TMS320C5x TMS320C6x TMS320C8x	-	TMS320C3x TMS320C4x
Motorola <sup>14</sup>	DSP56F80x/2x DSP5685x	DSP5600x DSP56002 DSP563x	DSP96002
NEC <sup>15</sup>	uPD77111 uPD77112 uPD77114	-	-
Zilog <sup>16</sup>	Z893x Z894X	-	-

**Tabla 1. 2:** Clasificación de los DSPs

<sup>12</sup> <http://www.analog.com/embedded-processing-dsp/processors/en/index.html>

<sup>13</sup> <http://focus.ti.com/dsp/docs/dsphome.tsp?sectionId=46>

<sup>14</sup> <http://www.freescale.com/>

<sup>15</sup> <http://www.necel.com/mobile/en/dsp/product/d7711x.html>

<sup>16</sup> [http://www.zilog.com/index.php?option=com\\_content&task=view&id=58&businessLine=1&parent\\_id=77&Itemid=185](http://www.zilog.com/index.php?option=com_content&task=view&id=58&businessLine=1&parent_id=77&Itemid=185)

## **1.3.2. FILTROS DIGITALES**

En las secciones precedentes se manifestó la importancia del procesamiento digital de señal en la actualidad, así como también se describió los elementos constitutivos de un sistema procesador digital de señal, resaltando entre estos la figura del microcontrolador DSP como componente principal, y se destacó su arquitectura dedicada a aplicaciones del procesamiento digital de señal, una de estas constituye el filtrado digital.

### **1.3.2.1. Técnicas de Filtrado digital**

Un filtro digital básicamente es un algoritmo matemático programable y expresable como una ecuación de diferencias, la primera característica se da por la posibilidad de determinar ciertos coeficientes que establecen la respuesta final de dicho filtro como pasa bajos, pasa altos, pasa banda, etc.

El filtrado digital es una parte del procesamiento digital de señal que ofrece muchas ventajas respecto del filtrado analógico, principalmente por su gran estabilidad en el tiempo, facilidades de tipo mecánico, mayor insensibilidad a condiciones ambientales, tipos de respuesta que no se pueden conseguir en filtros analógicos (como por ejemplo fase lineal), facilidades de diseño e implementación, etc.

Sin embargo no se puede afirmar de manera determinante que un filtro digital es mejor que un analógico, este último adjetivo es relativo a la aplicación en la cual se necesite aplicar filtrado, ya que existen condiciones en las cuales un filtro analógico se desenvuelve de mejor manera, como en potencias y frecuencias altas.

La finalidad de un filtro es procesar una señal presente en su entrada, de manera que la señal de salida presente ciertas características de frecuencia que obedezcan a las respectivas especificaciones de diseño.

#### ***1.3.2.1.1. Filtros FIR e IIR***

Existen muchas formas de clasificar los diferentes tipos de filtros digitales, quizás la más conocida, obedece al tipo de respuesta que se obtiene ante una entrada

unitaria, según esto pueden ser FIR (Filtros de respuesta finita) ó IIR (Filtros de respuesta Infinita).

Se puede mencionar mucho a cerca de cada uno de estos tipos de filtros digitales, aquí solo se mencionarán algunas características que determinen el uso de uno u otro tipo en el presente proyecto de titulación.

#### ***1.3.2.1.1.1. IIR***

La salida de estos filtros es una combinación lineal de los valores presentes y pasados de la señal de entrada, así como también de valores pasados de la salida, por lo que se les suele llamar recursivos. Las características más notables de este tipo de filtros digitales son:

- Para una determinada tarea pueden ser mucho más efectivos que los FIR y con un orden<sup>17</sup> más bajo, lo que redundo en una menor carga computacional.
- En principio no pueden diseñarse para tener fase lineal, es decir pueden distorsionar la respuesta de fase de la señal de salida, aunque pueden aplicarse ciertas técnicas para lograrlo, como el llamado *filtrado bidireccional*.
- Pueden resultar inestables, aun cuando se los diseñe para ser estables.

#### ***1.3.2.1.1.2. FIR***

La salida de estos filtros es una combinación lineal de los valores presentes y pasados de la señal de entrada, por lo que se les suele llamar no recursivos. Las características más notables de este tipo de filtros digitales son:

- Para una determinada tarea necesitan de un orden más alto que los IIR, lo que redundo en una mayor carga computacional.
- Pueden ser diseñados para ser de fase lineal, para no distorsionar la fase de la señal de salida.

---

<sup>17</sup> El orden de un filtro digital determina la precisión del mismo, así mismo el número de coeficientes necesarios y el trabajo computacional que deberá realizar.

- Son siempre estables.

Según las características mencionadas anteriormente para cada uno de los tipos de filtros digitales, es más que suficiente el uso de filtros FIR, principalmente por su estabilidad y por la respuesta de fase lineal más fácil de conseguir que en los filtros IIR.

### 1.3.2.2. Diseño y simulación utilizando *Matlab*

El diseño de filtros digitales se ve facilitado por la existencia de software que ofrece un entorno GUI<sup>18</sup> intuitivo, donde se pueden introducir las especificaciones necesarias para obtener la respuesta deseada de un filtro; tal es el caso de *Matlab* que contiene una interfaz de usuario agradable para el diseño y simulación de todo tipo de filtros.

Esta herramienta se denomina ***FDATool***, para abrir la aplicación se escribe en el *command window* del *Matlab*, *fdatool*, a continuación se despliega en pantalla una ventana con todas las opciones para obtener la respuesta de frecuencia deseada, así tenemos los típicos y bien conocidos pasa bajos, pasabanda, pasa altos y elimina banda, pero también existen otros tipos de respuesta que se pueden diseñar tales como: filtros de coseno levantado, filtros diferenciadores, filtros que implementan la transformada *Hilbert*, entre otros.

La aplicación es muy didáctica y práctica, por cuanto para el diseño de un filtro en específico se pide introducir una serie de parámetros propios para cada tipo de filtro, entre lo más conocido está: la(s) frecuencia(s) de corte y las ganancias para las bandas de paso y de parada.

---

<sup>18</sup> *Graphical User Interface*

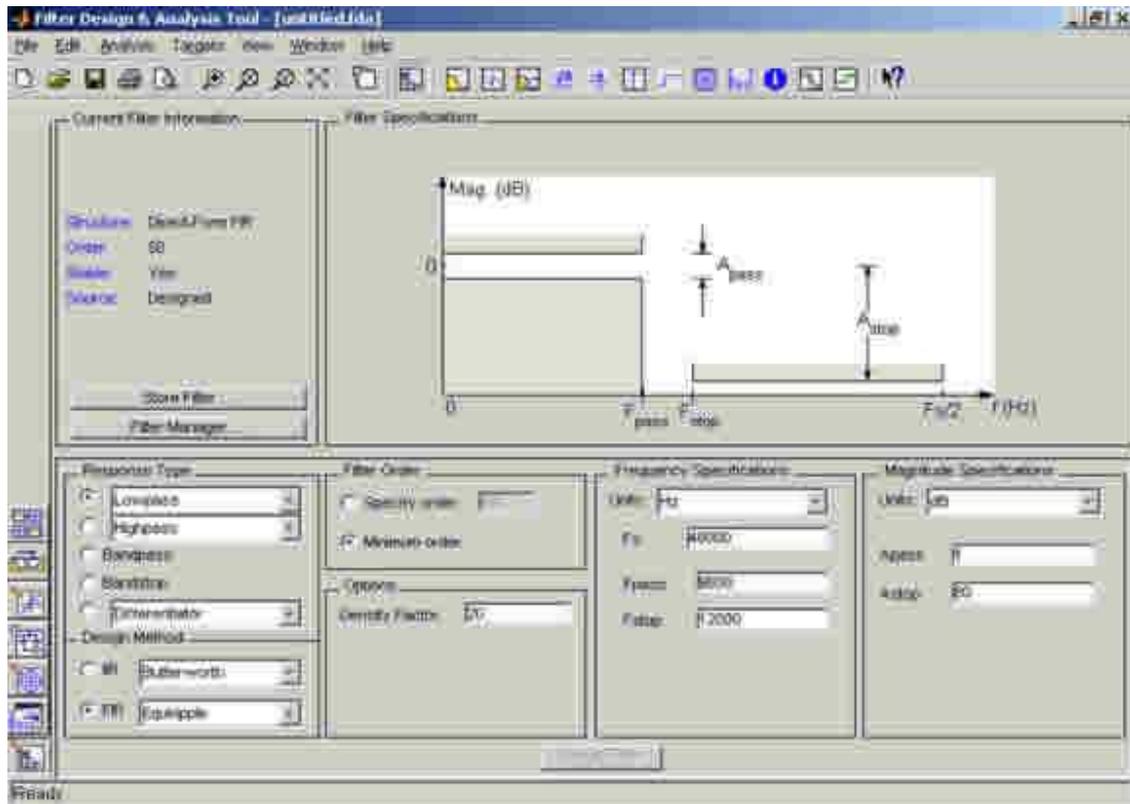


Figura 1. 16: FDATool de *MatLab*

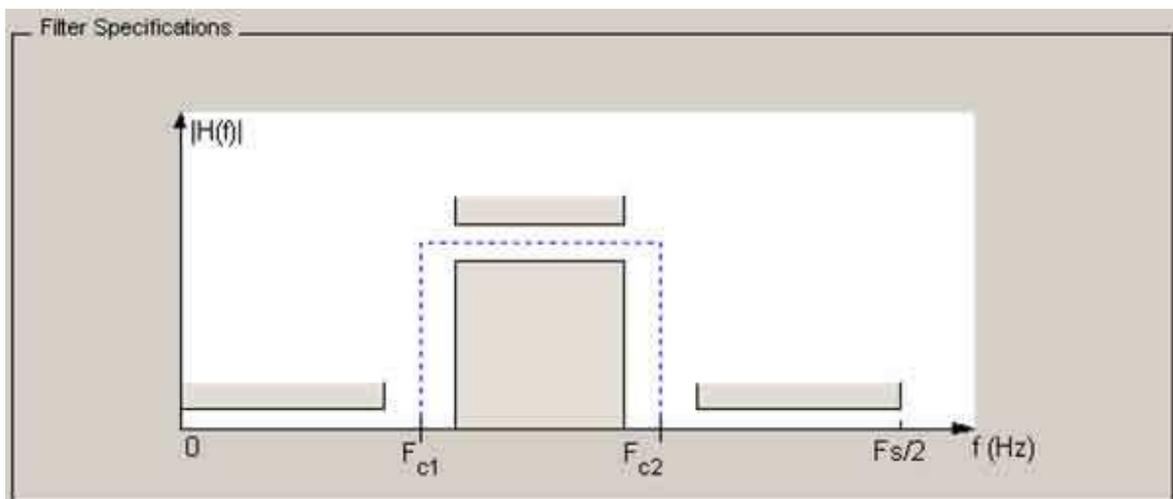
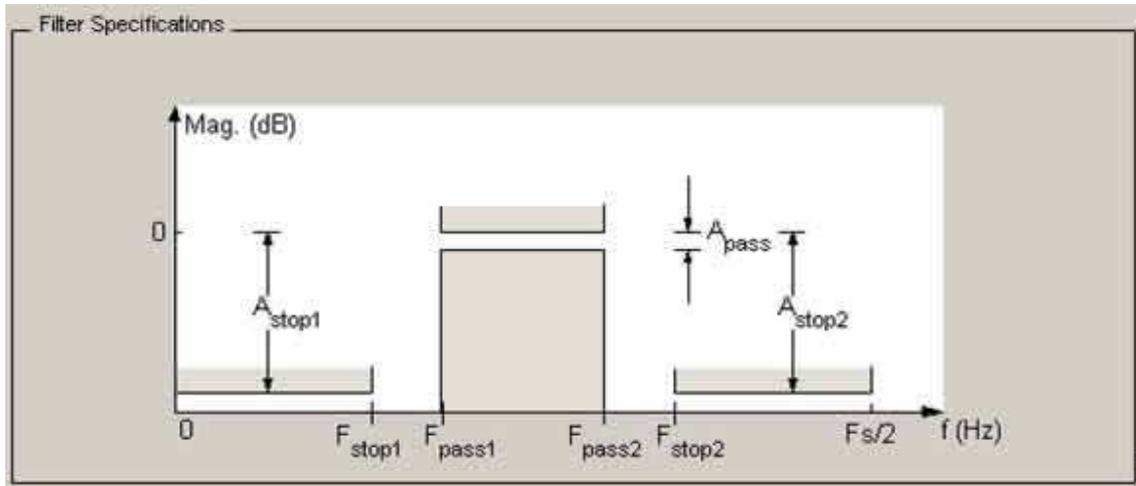
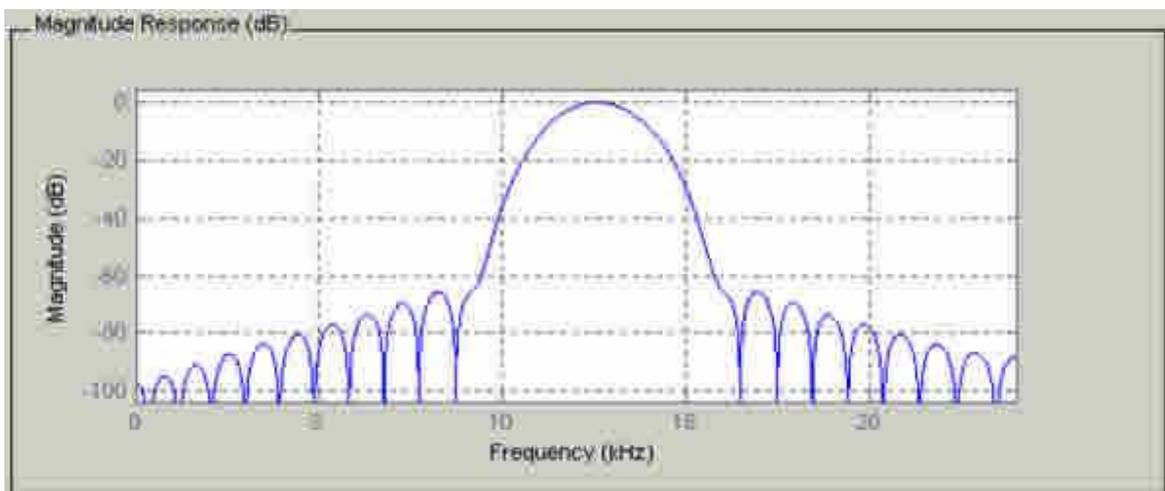


Figura 1. 17: Especificaciones para un filtro pasa banda normal



**Figura 1. 18:** Especificaciones para un filtro pasa banda más elaborado

Adicionalmente se tiene graficadores que hacen posible un análisis detenido de la respuesta del filtro deseado, en cuanto a frecuencia y fase:



**Figura 1. 19:** Respuesta en frecuencia de un filtro pasa banda

Existe adicionalmente una muy útil opción de exportar los coeficientes de los filtros generados hacia documentos en formato de texto plano.

### 1.3.2.2.1. Ejemplo de realización de un filtro digital con FDATAOOL

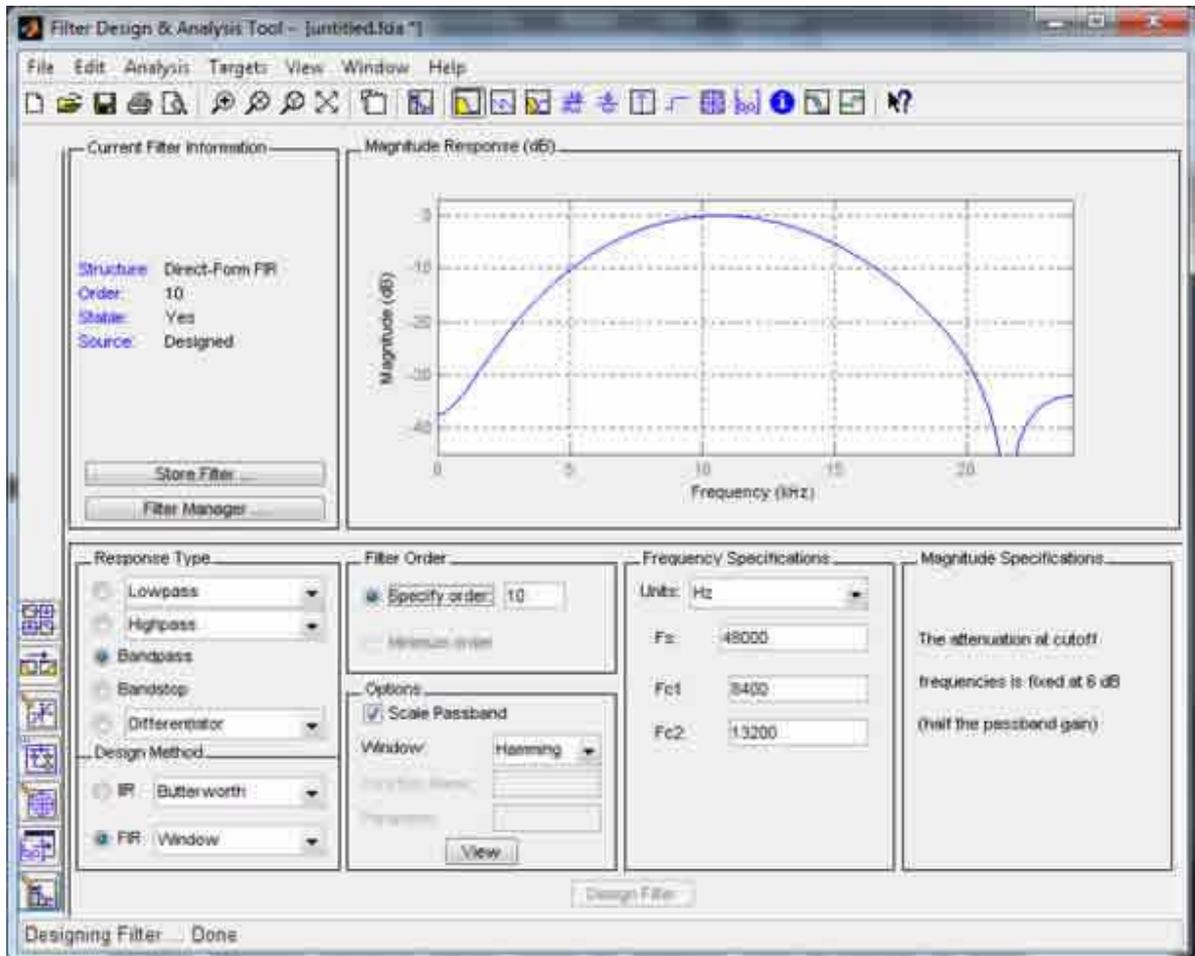
A continuación se establecerán los pasos necesarios para el diseño y simulación de un filtro digital con el objeto de obtener los coeficientes que determinarán la respuesta de frecuencia deseada según las siguientes especificaciones:

- Método de Diseño: *FIR*
- Tipo de respuesta: *Pasabanda*
- Orden del filtro: 10
- $F_{c1} = 8400 \text{ Hz}$
- $F_{c2} = 13200 \text{ Hz}$
- $F_s = 48000 \text{ Hz}$

Se debe entonces:

- Una vez abierta la aplicación *MatLab* se escribe en el *command Window* la palabra *fdatool*, con lo cual aparecerá la ventana que se muestra en la figura 1.16.
- Se escoge el tipo de respuesta en la sección *Response Type* como *Bandpass* (Pasabanda).
- El método de diseño en la sección *Design Method* como *FIR*, aquí se puede escoger la técnica con la que se lo hace, una típica manera es por ventaneo, así que se escoge *window*.
- El orden del filtro en la sección *Filter Order* donde se especifica sea de 10.
- En *Options* escogemos el tipo de ventana como *Hamming* ya que es una de las más utilizadas.
- En *Frequency Specifications* introducimos las especificaciones dadas anteriormente.
- Presionamos el botón *Design Filter*.
- Una vez realizado esto aparece una sección llamada *Magnitude Response (dB)* donde se grafica la respuesta de frecuencia del filtro diseñado, donde se puede comprobar el cumplimiento de las especificaciones con lo cual se

puede determinar o no la necesidad de una modificación. La figura siguiente muestra la ventana de *fdatool* correspondiente para el diseño de ejemplo:



**Figura 1. 20:** Ventana *FDATool* para el diseño del filtro de ejemplo

De la figura 1.20 se puede destacar una sección adicional identificada como *Current Filter Information*, donde se presenta un resumen de las características del filtro diseñado incluyendo además si es estable o no. En el Menú *Analysis* se puede acceder a los gráficos de respuesta de fase, de impulso, de escalón unitario, polos y ceros, etc.

- Ahora lo interesante es obtener los coeficientes del filtro para poder implementarlo en tiempo real a través de un sistema DSP, lo que se hace

entonces es sacar dichos coeficientes en un archivo en texto plano, lo cual se hace seleccionando del menú *File* la opción *Export* apareciendo la siguiente ventana:



**Figura 1. 21:** Opción Exportar Coeficientes de *FDATool*

Simplemente se hace clic en *export* apareciendo un cuadro de diálogo en donde se escoge el nombre y destino del archivo en texto plano que contiene los coeficientes, el contenido de dicho archivo se muestra a continuación:

```
%
% Generated by MATLAB(R) 7.6 and the Signal Processing Toolbox 6.9.
%
% Generated on: 01-Oct-2009 20:26:59
%

% Coefficient Format: Decimal

% Discrete-Time FIR Filter (real)
% -----
% Filter Structure   : Direct-Form FIR
% Filter Length     : 11
% Stable             : Yes
% Linear Phase      : Yes (Type 1)

Numerator:
 0.014087692242024141
 0.040203683211273954
-0.06065125200839265
-0.2374151231919075
 0.054905379720129543
 0.39118729687571868
 0.054905379720129543
-0.2374151231919075
-0.06065125200839265
 0.040203683211273954
```

0.014087692242024141

Se concluye entonces la facilidad en el diseño de filtros digitales utilizando *MatLab* y su herramienta *fdatool* tornándose incluso intuitiva su utilización con un mínimo de conocimiento en la materia.

Un procedimiento similar será el empleado en el diseño de los filtros necesarios en la implementación del *Firmware* que forma parte del sistema básico de localización vehicular.

### **1.3.3. GENERACIÓN DE FORMAS DE ONDA**

La generación de señales es un aspecto importante en el procesamiento digital de señales, ejemplos prácticos vienen dados en aspectos como el manejo de audio donde para lograr efectos de sonido tales como retardo, trémolo, phaser, flanger, vibrato, etc. es necesario implementar diversas formas de onda; en telecomunicaciones para generar la onda portadora que hará posible el envío de información entre lugares remotos, es precisamente para esto último en que será de utilidad los siguientes conceptos.

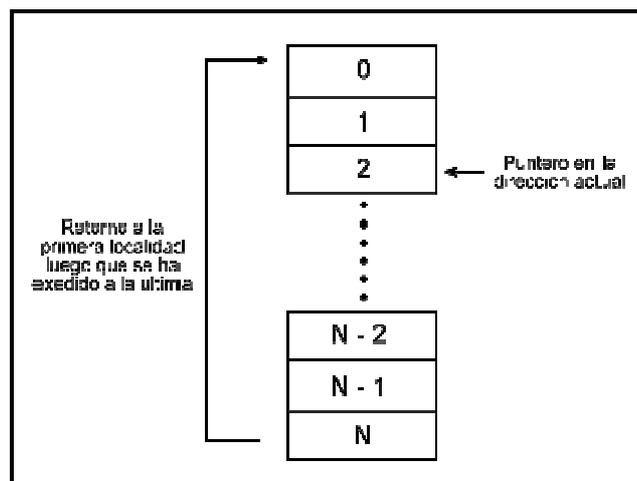
Existen varias maneras de generar una forma de onda utilizando un sistema DSP, las principales son:

- Programando en el microcontrolador DSP una subrutina que implemente una función de aproximación trigonométrica por series de *Taylor*.
- Usando una tabla que contenga las muestras necesarias para la forma de onda que se desea generar.

De las opciones mencionadas anteriormente se establece la segunda como la más idónea por considerarla de fácil implementación tomando en cuenta las facilidades prestadas por los DSPs; cómo se lo hace se explica a continuación:

En la sección 1.3.1.2.<sup>19</sup> se mencionó que una de las novedades en la arquitectura de los microcontroladores DSPs es la inclusión de estructuras de hardware llamadas DAGs, las cuales manejan el direccionamiento indirecto, mediante estas se puede implementar esquemas de almacenamiento circular.

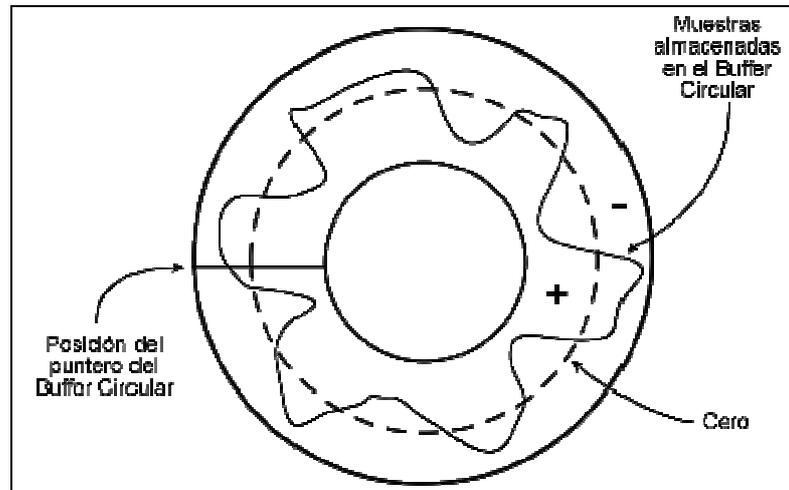
Un buffer circular es un segmento finito de localidades de memoria del DSP definido por el programador y usado para guardar *muestras* empleadas en el procesamiento. Una característica importante de los DAGs es su capacidad automática de incremento de los punteros para el acceso a las siguientes localidades de memoria cuando se ha realizado una lectura/escritura en dicho buffer, el calificativo de circular viene dado por una particularidad de esta capacidad, cuando se ha accedido a la última localidad de memoria perteneciente al buffer, el puntero de acceso regresa automáticamente a la primera localidad.



**Figura 1. 22:** Buffer circular de N localidades de memoria

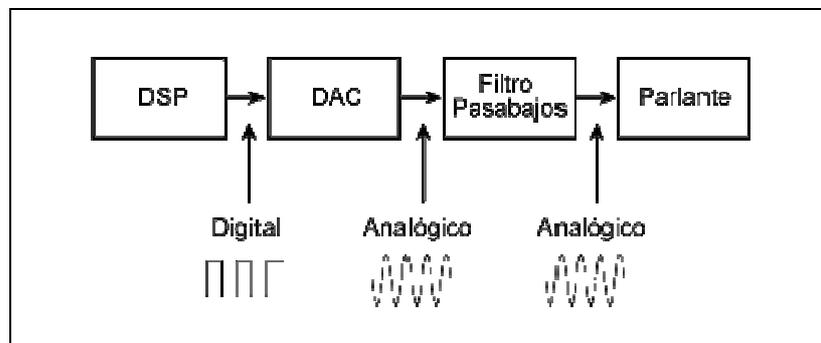
Se puede aprovechar entonces estas características para almacenar en un buffer circular las N muestras correspondientes a una forma de onda en particular, para generarla simplemente se realiza la lectura de una muestra cada cierto intervalo definido de tiempo, de modo que en un tiempo igual al período de la onda se hayan leído todas sus muestras, la capacidad explicada de los DAGs asegura que la onda se repita una y otra vez sin preocuparse cada vez de la posición del puntero.

<sup>19</sup> Página 23 del presente proyecto de titulación



**Figura 1. 23:** Forma de onda almacenada en un buffer circular

Luego de leída la muestra correspondiente se la lleva al convertidor digital – análogo (*DAC*) y así sucesivamente se lo hace con todas las muestras, finalmente todas estas atravesarán un filtro pasa bajos que suaviza el resultando final, obteniéndose así la forma de onda programada.



**Figura 1. 24:** Generación de formas de onda usando un DSP

Líneas arriba se mencionó que para generar la onda se debe realizar la lectura de la correspondiente muestra cada cierto intervalo de tiempo, de modo que se complete esta lectura de muestras en un tiempo igual al período de la onda a generar, entonces es necesario contar con una base de tiempo que permita establecer el instante preciso en el que se deba realizar la lectura de la muestra, en este sentido los microcontroladores DSP ofrecen típicamente dos opciones:

- Cada cierto tiempo los microcontroladores DSP realizan una interrupción de sus actividades para tomar una muestra proveniente del ADC, este tiempo es el período de muestreo de dicho ADC, se puede entonces aprovechar este instante para realizar la lectura de la muestra correspondiente y llevarla a la entrada del DAC.
- Otra opción es programar un timer del microcontrolador DSP para que cada cierto tiempo se realice la lectura de muestras.

La primera opción es viable en situaciones en que la frecuencia de onda a generar es múltiplo de la frecuencia de muestreo, caso contrario el número de muestras resulta no entera como veremos más adelante, es en estos casos en que se usa la segunda opción señalada.

Entonces, según la opción primera, como la frecuencia de muestreo es fija, el número de muestras requeridas para reconstruir la onda deseada dependerá de dicha frecuencia y se relaciona mediante la siguiente expresión:

$$N = \frac{f_s}{f_d} \quad (1.1)$$

Donde:

$f_s$  = frecuencia de muestreo

$f_d$  = frecuencia de la forma de onda deseada

Es evidente que la igualdad anterior debe ser un número entero y que este no puede ser inferior a dos por el teorema de Nyquist

Todo lo señalado debe ser coherente, por ejemplo:

Usando una frecuencia de muestreo  $f_s$  de 48000 Hz, la máxima frecuencia que se puede reconstruir a partir de muestras es de 24000 Hz; supongamos que se desea generar una onda de frecuencia 2400 Hz, según la expresión (1.1), N resulta ser de

20, este número de muestras debe ser leído completamente en un tiempo igual al período de la onda a generar, es decir en:

$$\frac{1}{2400} s = 0.00041666 s$$

Como se toma como base de tiempo el período de muestreo se tiene que se lee una muestra cada:

$$\frac{1}{48000} s = 20.83 \mu s$$

Entonces la totalidad de muestras se lee en:

$$20 * 20.83 \mu s = 0.00041666 s$$

Que es igual al período de la onda que se desea generar

Así, el número de localidades de memoria a reservar en el buffer circular es idéntico al número de muestras necesarias para reconstruir la forma de onda a partir de las mismas, para el ejemplo dado resultó ser de 20.

La frecuencia de onda máxima que se puede generar viene dada por la máxima frecuencia de muestreo que pueden ofrecer los convertidores *ADC – DAC* utilizados en el sistema DSP, según el teorema de Nyquist:

$$f_s \geq 2f_m \quad (1.2)$$

Dónde:

$f_s$  = frecuencia de muestreo

$f_m$  = máxima frecuencia en la señal muestreada

Así por ejemplo, si la máxima frecuencia de muestreo en un sistema DSP es de 24000 Hz, no se podrá generar una forma de onda con una frecuencia mayor a 12000 Hz.

Ahora, si la frecuencia que se desea generar no es múltiplo de la frecuencia de muestreo se deberá usar el timer del microcontrolador DSP para obtener mayor precisión, se usa entonces una versión modificada de la expresión (1.1):

$$f_l = M * f_d \quad (1.3)$$

Dónde:

$f_l$  = Es la frecuencia con que se leerán las muestras de la onda a generar

$f_d$  = frecuencia de la forma de onda deseada

$M \geq 2$  representa el número de muestras para reproducir la forma de onda deseada

Por ejemplo se desea generar una frecuencia de 3600 Hz teniendo como frecuencia de muestreo 48000 Hz, evidentemente no se puede usar la expresión (1.1), se usa entonces la expresión (1.3) escogiendo arbitrariamente para M el valor de 6, resultando tener  $f_l$  un valor de 21600 Hz, entonces el valor a programar en el timer es de:

$$\frac{1}{21600} s = 46.296 \text{ us}$$

Entonces tomando como base de tiempo el programado en el timer, las seis muestras deben leerse en un tiempo de:

$$6 * 46.296 \text{ us} = 277.77 \text{ us} = \frac{1}{3600} s$$

Es así entonces como se genera una forma de onda mediante una tabla utilizando un microcontrolador DSP, adicionalmente se debe decir que las muestras de las ondas que se desean generar pueden ser obtenidas en *MatLab*.

## 1.4. EZ KIT LITE ADSP – 2181

### 1.4.1. DESCRIPCIÓN GENERAL

El Ez Kit Lite de *Analog Devices* es la herramienta disponible con la cual se va a trabajar, representa por sí misma una aplicación particular de lo que se puede hacer con el procesador digital de señal ADSP – 2181, y constituye una plataforma para el desarrollo de otras aplicaciones que requieren hacer uso, aparte de las capacidades de procesamiento digital del 2181, de características de entrada y salida (conexión serial a la PC, entrada y salida de señales de audiofrecuencia, banderas de usuario, etc.) para interactuar con el exterior obteniendo e introduciendo datos.

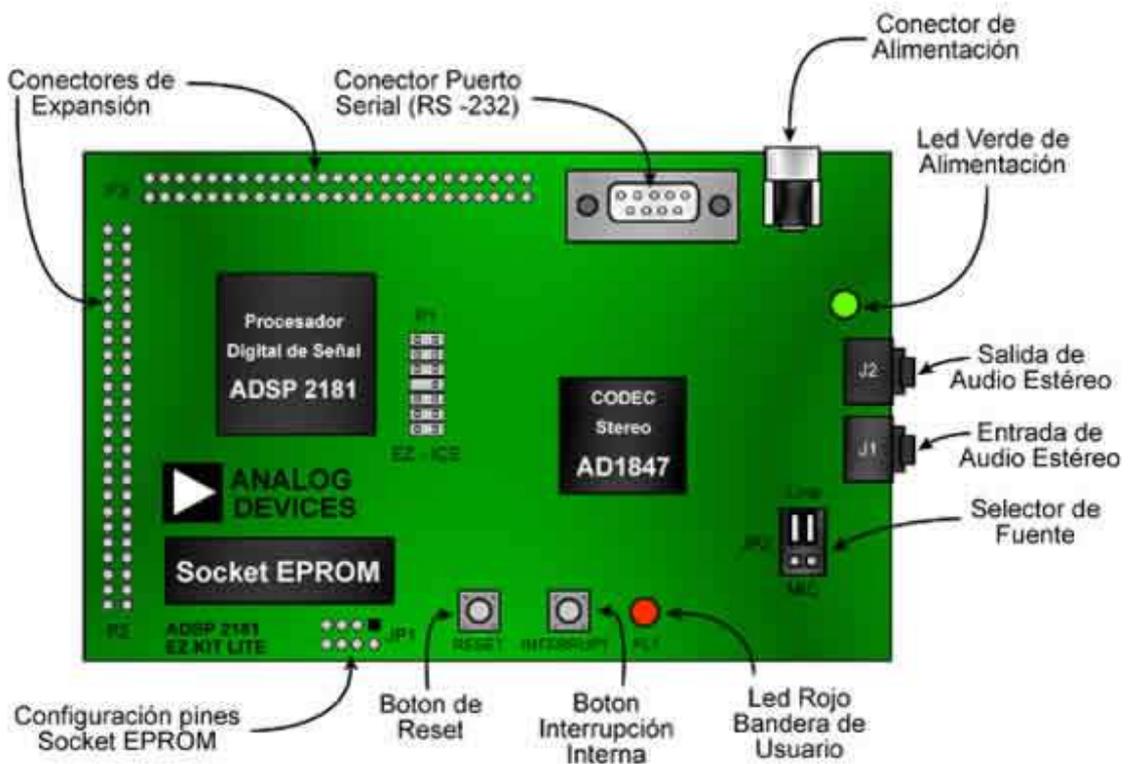


Figura 1. 25: Tarjeta de desarrollo Ez Kit Lite ADSP 2181

## **1.4.2. ARQUITECTURA BASE**

A continuación se hace una breve descripción de las partes constitutivas fundamentales de la tarjeta de desarrollo.

### **1.4.2.1. Microcontrolador ADSP 2181**

Constituye el núcleo del Ez Kit siendo parte esencial del sistema DSP y con el cual se realizará todo el procesamiento digital de señal necesario en el presente proyecto de titulación, este microcontrolador está optimizado para realizar operaciones numéricas de alta velocidad exigidas en aplicaciones de procesamiento digital de señal, está diseñado para realizar 33 MIPS (*Millones de instrucciones por segundo*), es decir cada instrucción tiene un tiempo de duración de 30 nano segundos, su arquitectura trae consigo diferencias importantes respecto a un micro común lo que le da la categoría de DSP (Procesador digital de señal); trabaja con 16 bits en punto fijo.

#### ***1.4.2.1.1. Unidades Computacionales***

Las unidades computacionales son la base del microcontrolador DSP ya que mediante ellas se lleva acabo todo el procesamiento de datos, cada una es independiente y realiza operaciones determinadas.

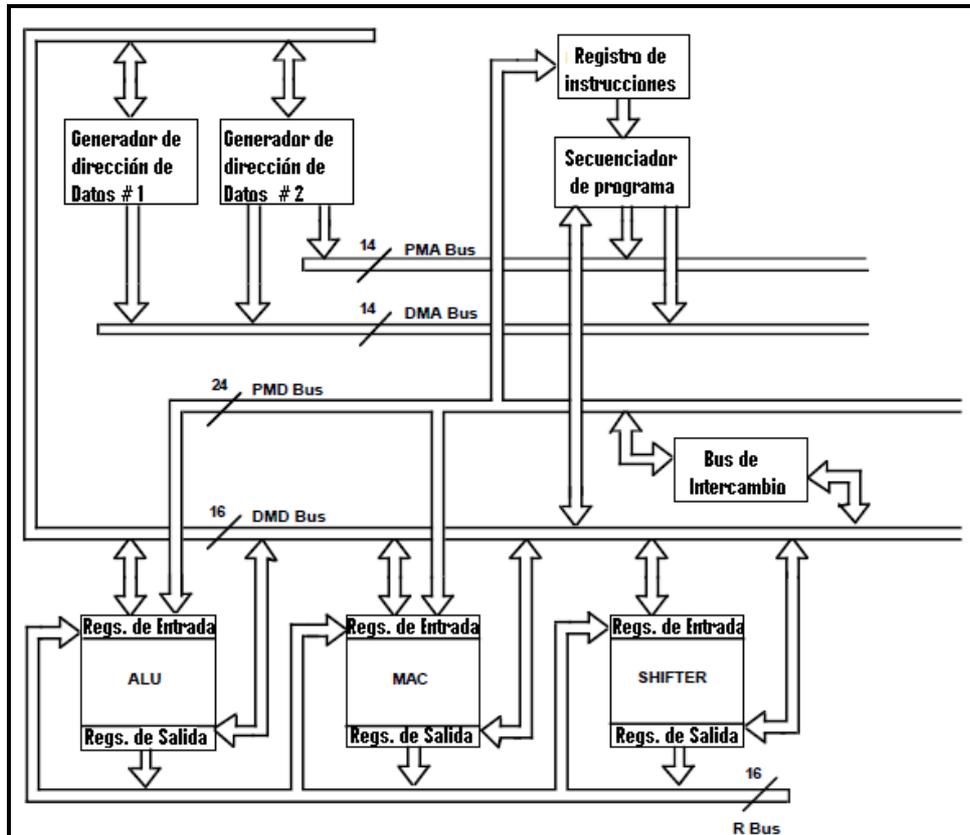


Figura 1. 26: Arquitectura del ADSP 2181

#### 1.4.2.1.1.1. Unidad aritmética Lógica (ALU)

La Unidad aritmética lógica (ALU) es la misma que se puede encontrar en cualquier micro y se encarga de realizar las operaciones aritméticas y lógicas básicas, entre las cuales se cuenta la suma, resta, ciertas divisiones primitivas, AND, OR, NOT y XOR (OR exclusivo) respectivamente. Estas operaciones están optimizadas para ser realizadas en formato binario 1.15, comúnmente estos formatos numéricos se designan como  $a.b^{20}$ . Tanto los operandos como los resultados se encuentran en 16 bits, excepto la división primitiva.

#### 1.4.2.1.1.2. Unidad de Multiplicación y Acumulación (MAC)

<sup>20</sup>  $a$  representa el número de dígitos binarios empleados para representar la parte entera, y  $b$  el número de dígitos binarios empleados para representar la parte fraccionaria, así  $a$  es igual a 1 y  $b$  igual a 15.

Nótese que siendo la multiplicación una operación aritmética básica no se la nombró en el apartado anterior, ya que esta es realizada mediante una unidad computacional totalmente independiente de la ALU, esta se la denomina unidad de multiplicación y acumulación, pues es capaz de realizar multiplicaciones de doble precisión e inclusive acumular (sumar) los resultados en un solo ciclo de máquina, es decir 30 nano segundos. Esta constituye una diferencia fundamental con los micros de arquitectura común, en los cuales las operaciones de multiplicación se deben implementar por software, mientras que en un procesador digital de señal estas son realizadas mediante hardware dedicado, lo cual es una ventaja en cuanto a rapidez de procesamiento se refiere. Los factores se encuentran en 16 bits pero el resultado esta en 32 bits ó inclusive hasta 40 bits si se producen desbordamientos.

#### ***1.4.2.1.1.3. Unidad de desplazamiento (SHIFTER)***

Esta unidad computacional proporciona un completo conjunto de funciones útiles en el tratamiento de datos, por ejemplo desplazamientos aritméticos y lógicos, normalización, derivación de exponente, entre otras, todo lo mencionado se realiza eficientemente en cualquier formato numérico del tipo a.b. Las entradas están en 16 bits y la salida puede darse hasta en 32 bits.

Para un mayor detalle en cuanto a la arquitectura de las unidades computacionales se puede consultar el manual de referencia de la familia de procesadores 2100 de *Analog Devices*<sup>21</sup>.

#### ***1.4.2.1.2. Control del programa***

##### ***1.4.2.1.2.1. Secuenciador de Programa***

El secuenciador de programa controla el flujo adecuado en el cumplimiento correcto de las instrucciones, este toma en cuenta la aparición de saltos, llamadas a subrutinas y da soporte para el servicio a interrupciones, tanto internas como

---

<sup>21</sup> ANALOG DEVICES, ADSP-2100 Family User's manual, Tercera Edición, Canadá, 1995, pp. 2-1 – 2-36

externas, de tal forma que se cumpla los objetivos del usuario programador especificados en el respectivo algoritmo.

#### *1.4.2.1.2.2. Interrupciones*

El conjunto de interrupciones a las cuales se puede dar servicio suman en total doce, de las cuales ocho son externas y la diferencia internas, a continuación se las enumera en orden descendente de prioridad:

<b>INTERRUPCIÓN</b>
<i>RESET (más alta prioridad)</i>
<i>Power Down (No enmascarable)</i>
<i>IRQ2</i>
<i>Escritura Hip</i>
<i>Lectura Hip</i>
<i>Transmisión SPORT0</i>
<i>Recepción SPORT0</i>
<i>Transmisión análoga (DAC)</i>
<i>Recepción análoga (ADC)</i>
<i>Transmisión SPORT1 ó IRQ1</i>
<i>Recepción SPORT1 ó IRQ0</i>
<i>Timer (más baja prioridad)</i>

**Tabla 1. 3:** Interrupciones en el ADSP 2181

El manejo de interrupciones hace necesario la inclusión de algunos registros que permitan configurar el desenvolvimiento de las mismas, entre los importantes se encuentra el de “enmascaramiento” que controla la habilitación individual de interrupciones.

Cada una de las interrupciones enumeradas en la tabla 1.2 tiene su propia dirección de inicio en la memoria de programa y hay un espacio de cuatro direcciones de dicha memoria antes de que comience una nueva dirección de programa correspondiente a otra interrupción, lo que permite dar servicio a rutinas pequeñas de interrupción sin que sea necesario realizar un salto, caso contrario se debe realizar uno para dar servicio a una rutina grande de interrupción (más de cuatro instrucciones). Para una

información más detallada se puede acudir al manual de referencia de la familia de procesadores 2100 de *Analog Devices*<sup>22</sup>.

#### ***1.4.2.1.3. Transferencia y almacenamiento de datos***

La transferencia hace referencia a la forma en que se acarrean los datos desde y hacia las diferentes unidades computacionales, integrando de esta manera los diferentes componentes de la arquitectura del procesador ADSP 2181 mostrada en la figura 1.24.

##### ***1.4.2.1.3.1. Tipos de Buses***

El procesador ADSP 2181 tiene cinco buses internos que se encargan de transferir los datos, los buses PMA (*Program Memory Adress*) y DMA (*Data Memory Adress*) son usados internamente para llevar las direcciones tanto de la memoria de programa como la de datos respectivamente, y cada uno está formado por 14 bits. Los buses PMD (*Program Memory Data*) y DMD (*Data Memory Data*) son usados internamente para llevar los datos tanto de la memoria de programa como de la memoria de datos y están formados por 24 y 16 bits respectivamente. El bus R (*Result*) es de 16 bits y contiene, como su nombre lo indica, los resultados de las diferentes unidades computacionales.

##### ***1.4.2.1.3.2. Generadores de direcciones de datos***

Conocidos como DAGs, el ADSP 2181 posee dos en total y son encargados del direccionamiento indirecto, posibilitando la rápida transferencia de datos, ya que el procesador no se encarga de traer primero la dirección donde se encuentra el dato y en el siguiente ciclo acceder al dato, esto sucede en los micros comunes, desperdiando así valioso tiempo que en aplicaciones de procesamiento digital de de señal de tiempo real no se puede perder. Por el contrario en la arquitectura del ADSP 2181 se implementa en hardware estos DAGs con lo cual el procesador ya no realiza la primera tarea mencionada anteriormente sino que, en un solo ciclo accede

---

<sup>22</sup> ANALOG DEVICES, ADSP-2100 Family User's manual, Tercera Edición, Canadá, 1995, pp. 3-11 – 3-19.

directamente al dato deseado pues el DAG utilizado genera la dirección donde se encuentra el mismo de forma automática. Esta constituye otra diferencia fundamental con los micros comunes.

#### ***1.4.2.1.3.3. Memoria***

El ADSP 2181 como todos los microcontroladores DSP utiliza una arquitectura de memoria Harvard, cuyas características y ventajas fueron ya explicadas.

La memoria interna de programa del *ADSP2181* tiene un tamaño de 16K x 24 bits y la memoria de datos de 16k x 16 bits.

#### **1.4.2.2. Periféricos Incorporados**

Los periféricos incorporados permiten interactuar con el usuario de tal manera que se le permita depurar el desarrollo de alguna aplicación DSP.

##### ***1.4.2.2.1. Puerto Serial***

El ADSP 2181 tiene dos puertos seriales designados como SPORT0 y SPORT1, los cuales pueden ser configurados para cumplir con los requerimientos del usuario proporcionando flexibilidad en la comunicación serial, pueden transmitir y recibir simultáneamente para una operación full dúplex, tienen capacidad de operación multicanal, cuando transmiten o reciben generan una interrupción, pueden usar un reloj externo o uno interno, el SPORT1 puede ser configurado como dos interrupciones externas (IRQ0 y IRQ1), Ambos SPORTs poseen Auto bufferado.

Típicamente el puerto serial se lo usa para dos tareas, la primera para obtener datos de alguna fuente serial, como por ejemplo una PC o un GPS, la segunda para transferir el programa hacia la memoria del DSP cuando se encuentra en etapa de depuración y la memoria EPROM solo contiene el programa monitor<sup>23</sup> y no el programa final.

---

<sup>23</sup> Se refiere al programa HOST que trae consigo el EZ KIT lite ADSP 2181

Para configurar los puertos seriales se debe setear algunos bits en los registros apropiados para tal efecto. Para una información más detallada se puede acudir al manual de referencia de la familia de procesadores 2100 de *Analog Devices*<sup>24</sup>

#### **1.4.2.2. Banderas entrada/salida**

La tarjeta de desarrollo trae consigo una bandera de salida de usuario denominada FL1, la cual se la usa típicamente para depurar el programa en desarrollo configurándola para activarse cuando ocurra algún evento de interés, también se le puede añadir dos banderas de usuario designadas como FL2 y FL0 a través de los puertos de expansión que se muestran en la figura 1.23, para lo cual se debe añadir circuitería externa adicional.

El ADSP 2181 también posee banderas de entrada-salida adicionales ubicadas, como FL0 y FL2, en los puertos de expansión de la tarjeta de desarrollo y son conocidas como PF, numeradas del 0 al 7; también se puede considerar como banderas de entrada y salida a la función alterna del SPORT1 conocidas como FLAG IN y FLAG OUT.

#### **1.4.2.3. Códec AD1847**

Constituye una parte fundamental del sistema DSP descrito en la figura 1.13, se trata de un códec de audio estéreo (dos canales), el cual se encarga de la conversión analógico – digital y digital – analógico, posee capacidad de atenuación y ganancia programables, frecuencias programables de muestreo desde los 5.5 KHz. hasta 48 KHz.

El códec es un elemento fundamental ya que permite que el ADSP 2181 pueda procesar digitalmente la información analógica externa y devolverla en formato analógico nuevamente.

---

<sup>24</sup> ANALOG DEVICES, ADSP-2100 Family User's manual, Tercera Edición, Canadá, 1995, pp. 5-1 – 5-42

### 1.4.3. SOFTWARE DE DESARROLLO

#### 1.4.3.1. Breve descripción del tipo de programación utilizado

El lenguaje de programación utilizado es el llamado *lenguaje Assembler*, de bajo nivel, este constituye la representación más directa del código de máquina, específico para la arquitectura del ADSP 2181, legible por un programador.

El lenguaje ensamblador implementa una representación simbólica de los códigos de máquina (unos y ceros) y otras constantes necesarias para un programa a ser ejecutado en una arquitectura específica, en nuestro caso la del ADSP 2181. Esta representación es generalmente definida por el fabricante del hardware (*Analog Devices*), y se basa en las abreviaturas (llamados mnemónicos) que ayudan a recordar al programador instrucciones, registros, etc.

#### 1.4.3.2. VisualDSP++ 3.5

Es el software computacional que facilita la redacción del código fuente, su traducción a lenguaje de máquina y fácil depuración, el Ez Kit lite ADSP 2181 incluye un paquete completo de desarrollo para el sistema operativo DOS (CLI<sup>25</sup>), pero actualmente existen herramientas modernas basadas en un entorno GUI<sup>26</sup> que facilitan la implementación de algoritmos de programación, y es precisamente esta opción la que utilizaremos a través del software de *Analog Devices Visual DSP++ 3.5*.

*Visual DSP++ 3.5* es una herramienta interesante para el desarrollo de aplicaciones DSP para microcontroladores DSP ADI, la versión 3.5 soporta los modelos 21xx e inclusive algunas funcionalidades de los microcontroladores mencionados adelante, una versión más reciente, la 5.0, permite interactuar con los modelos *Blackfin*, *Sharc* y *Tigersharc*. La interfaz amigable con el programador de *Visual DSP++ 3.5* es como se muestra en la figura siguiente:

---

<sup>25</sup> Interfaz de línea de comando

<sup>26</sup> Interfaz Gráfica de usuario

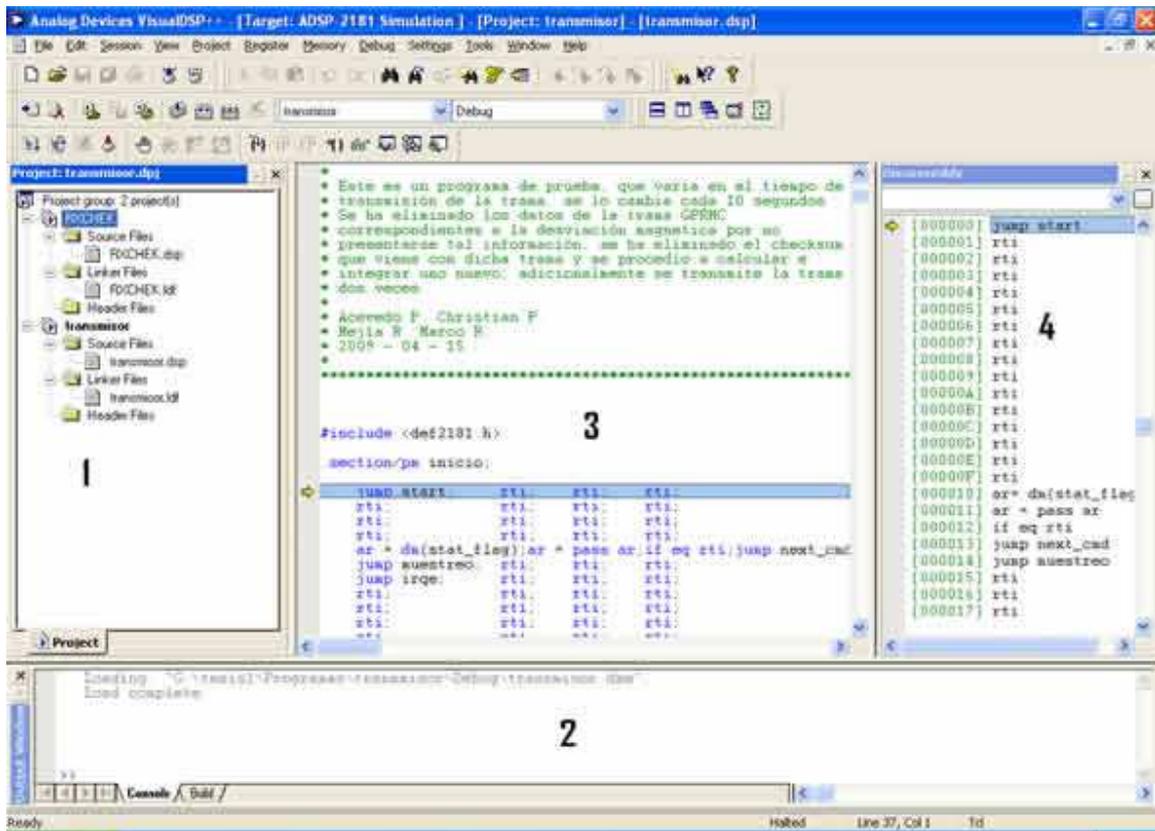


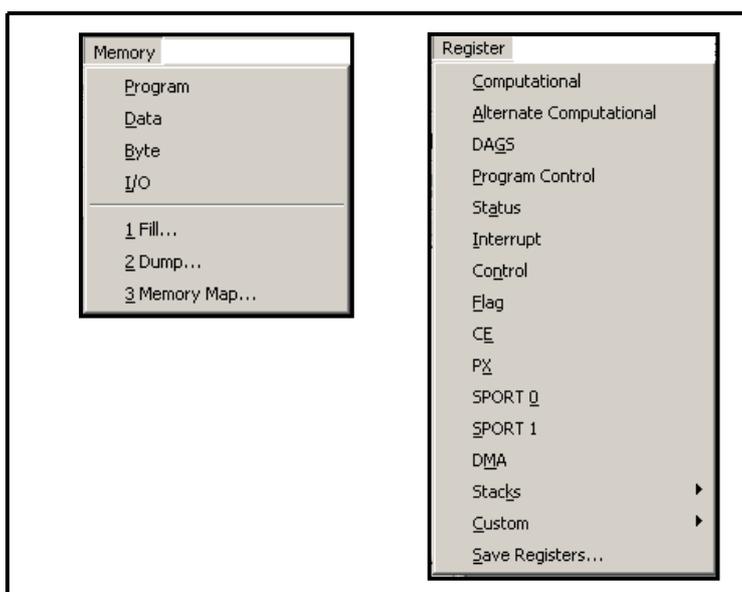
Figura 1. 27: Entorno de trabajo del Visual DSP++ 3.5

La figura 1.25 destaca el entorno de trabajo normal del Visual DSP++ 3.5 donde se aprecian:

1. **Ventana de proyecto:** en esta se enumeran los diferentes proyectos en los cuales se está trabajando, denotando con negrita el proyecto activo; cada proyecto tiene al menos dos carpetas dentro de las cuales se halla el archivo fuente y el archivo de enlace, este último contiene la declaración y extensión de las diferentes secciones de memoria (Datos o programa) en el archivo fuente.
2. **Ventana de salida (Output Window):** En esta se presentan todos los sucesos que originan procesos como el de compilado, por ejemplo de éxito o en su defecto de error o advertencia.

3. **Ventana de desarrollo:** En esta se realiza la redacción del código fuente, Visual DSP++ identifica los mnemónicos de registros e instrucciones y los diferencia coloreándolos de color azul, los comentarios de color verde, y el resto de texto, como variables y constantes de color negro.
4. **Ventana de desmontaje (*Disassembly Window*):** Muestra el flujo del programa instrucción a instrucción.

En *Visual DSP++* también se pueden hacer algunas tareas de simulación, permitido por la inclusión de ventanas adicionales similares a la de desmontaje, relacionadas con la memoria de datos, memoria de programa, control de programa, etc. Estas se las puede encontrar y seleccionar en los siguientes menús:



**Figura 1. 28:** Menús importantes del Visual DSP++ 3.5

#### ***1.4.3.2.1. Puesta en marcha de Visual DSP++ 3.5***

Para su utilización *Visual DSP++* debe ser adquirido a uno de los representantes oficiales de *Analog Devices* en Sudamérica ubicados en Venezuela, Brasil o

Argentina, o bien se puede conseguir una copia de evaluación completamente operativa por 90 días en la siguiente dirección:

[http://forms.analog.com/Form\\_Pages/processors/visualDSPTestDrive.asp](http://forms.analog.com/Form_Pages/processors/visualDSPTestDrive.asp)

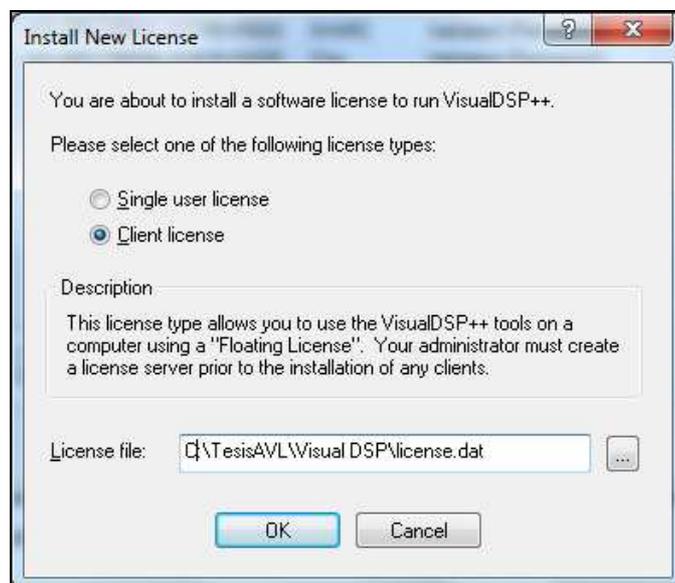
Los requerimientos mínimos del sistema son:

*Microsoft Windows 2000, XP SP2, Windows vista.*

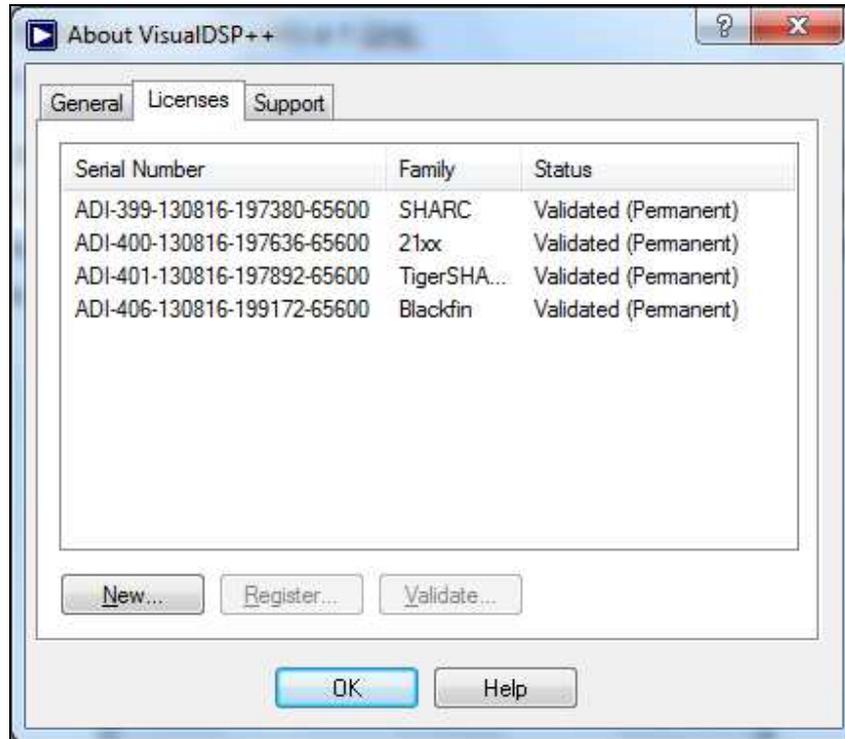
Procesadores Intel (x86) a 1 GHz.

512 MB RAM y 2 GB de espacio libre en disco.

En el caso de adquirir el producto habrá que registrarlo en *Analog Devices* para poder obtener las licencias de utilización para cada modelo de microcontrolador DSP ADI, una vez instalado *VisualDSP++*, en el primer inicio del mismo se pide dichas licencias, en este caso se proveerá de una de tipo cliente:

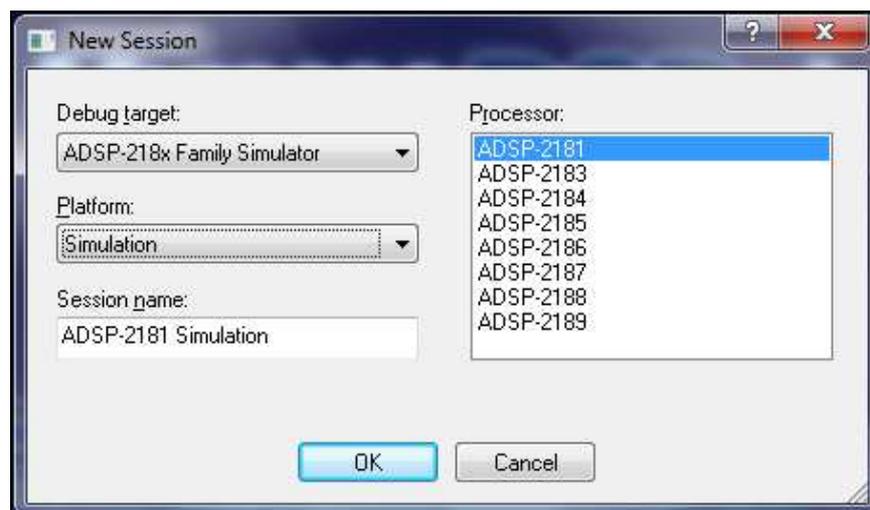


**Figura 1. 29:** Instalación de licencias en *VisualDSP++ 3.5*



**Figura 1. 30:** Licencias instaladas en VisualDSP++ 3.5

A continuación se debe crear una *Nueva Sesión* fijando las opciones según corresponda, en este caso son las que muestra la siguiente imagen:



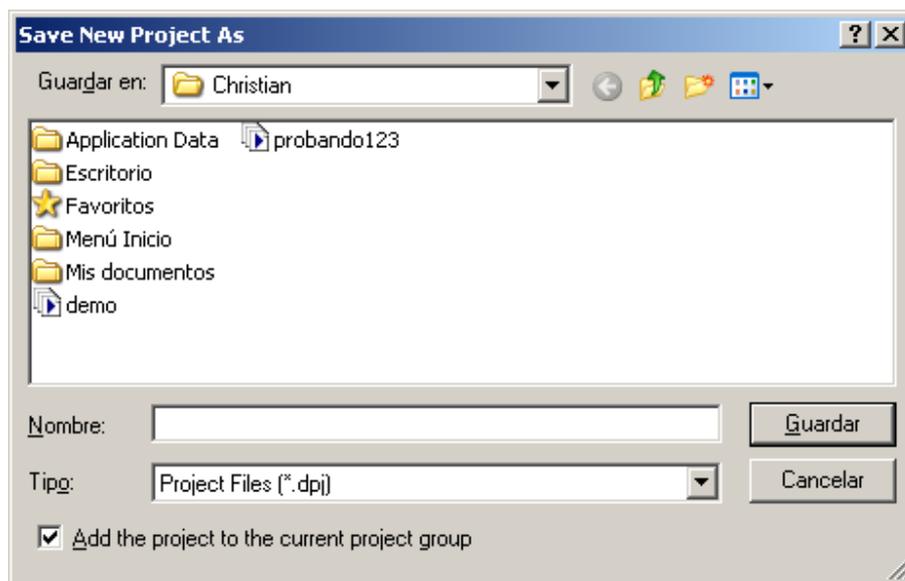
**Figura 1. 31:** Creación de una nueva sesión en VisualDSP++ 3.5

Luego de esto se inicia la interfaz de usuario mostrada en la figura 1.25 donde se puede comenzar a trabajar.

#### 1.4.3.2.2. Ejemplo práctico de Utilización de VisualDSP++ 3.5

A continuación se realizará el “Hello World” de VisualDSP++ 3.5, se explicará entonces la implementación de un programa con la funcionalidad sencilla de prender y apagar un *led* (Bandera de salida FL1) con una frecuencia visible. Los pasos a seguir para esto son los siguientes:

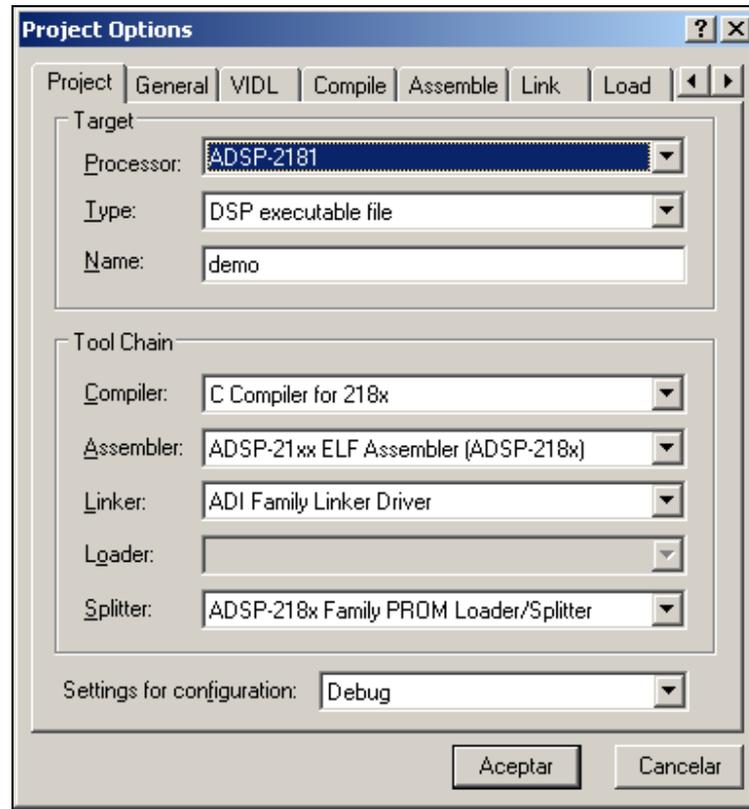
- En el entorno de Visual DSP++ hacemos clic en *Project* y en *New*, mostrándose la siguiente pantalla:



**Figura 1. 32:** Cuadro de diálogo nuevo proyecto

Donde se escribe el nombre del proyecto, con el cual se lo identificará a lo largo del desarrollo de la aplicación DSP.

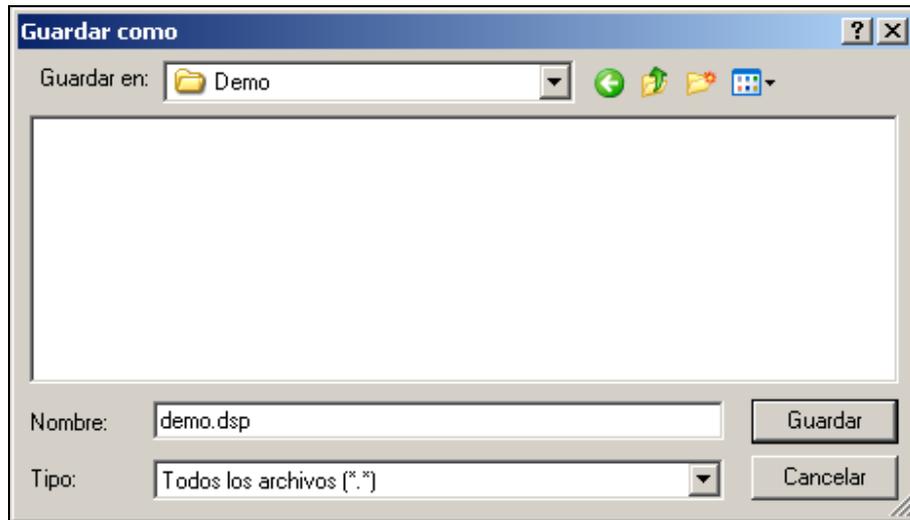
- A continuación se muestra las opciones del Proyecto:



**Figura 1. 33:** Opciones de configuración de nuevo proyecto

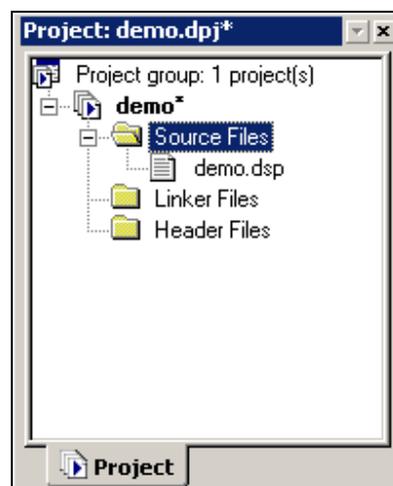
Por lo pronto se dirá que las opciones por defecto son suficientes, pero cabe recalcar que hay una serie de parámetros que se pueden establecer en este cuadro de diálogo, a través de las diferentes pestañas disponibles.

- Crear un nuevo archivo escogiendo del menú *File*, la opción *New* con lo cual aparecerá el ventana de desarrollo descrita anteriormente, en la cual se podrá escribir el archivo fuente, pero antes de esto se debe guardar el archivo en la opción *Save As* del menú *File* donde guardaremos nuestro archivo fuente con cualquier nombre, pero con la extensión *.dsp* de la siguiente manera:



**Figura 1. 34:** Cuadro de diálogo guardar proyecto

- Una vez realizado el paso anterior se debe agregar el archivo fuente a la carpeta llamada *Source Files* en la ventana de proyecto, esto se logra dando un clic derecho en dicha carpeta y escogiendo la opción *Add File(s) to Folder* en donde se visualizará un cuadro de diálogo donde encontraremos el archivo con extensión *.dsp* guardado anteriormente.



**Figura 1. 35:** Archivos fuente del proyecto

- Ahora se tendrá que escribir dentro del archivo *demo.dsp* el código fuente para la aplicación mencionada, el cual estará escrito en lenguaje *assembler*, dicho código es el siguiente:

```

/*Programa de demostracion del Visual DSP++ 3.5*/

#include <def2181.h>;          /* Libreria especifica para la arquitectura del
ADSP 2181 */

.section/pm interrupciones;    /* Puntero de direccion de la seccion de
                               interrupciones en memoria de programa */

    jump start; rti;rti;rti; /*00: reset */
    rti;        rti;rti;rti; /*04: IRQ2 */
    rti;        rti;rti;rti; /*08: IRQL1 */
    rti;        rti;rti;rti; /*0c: IRQL0 */
    rti;        rti;rti;rti; /*10: SPORT0 tx */
    rti;        rti;rti;rti; /*14: SPORT0 rx */
    rti;        rti;rti;rti; /*18: IRQE */
    rti;        rti;rti;rti; /*1c: BDMA */
    rti;        rti;rti;rti; /*20: SPORT1 tx or IRQ1 */
    rti;        rti;rti;rti; /*24: SPORT1 rx or IRQ0 */
    rti;        rti;rti;rti; /*28: timer */
    rti;        rti;rti;rti; /*2c: power down */

.section/dm variables;        /* Puntero de dirección de la sección de
                               variables en memoria de datos*/
.var retardo = 0xFFFF;       /* Variable que contiene el valor de 65535
                               para generar un retardo*/
.var retardo_1 = 0x007E;     /* Variable que contiene el valor de 126 para
                               generar un retardo*/

.section/pm programa;        /* Puntero de dirección de la sección de
                               código en memoria de programa */
start:
                               /* Primera linea de programa */
    cntr = dm(retardo_1);     /* Definición del primer contador para la
                               generación del retardo */
    do leds until ce;         /* Loop que realiza tantas operaciones como el
                               primer contador lo especifique */
    cntr = dm(retardo);      /* Definición del segundo contador para la
                               generación del retardo */
    do leds_1 until ce;      /* Loop que realiza tantas operaciones como el
                               segundo contador lo especifique */
        leds_1: nop;         /* operación que realizara el loop do*/
        leds: nop;          /* operación que realizara el loop do*/
    toggle fl1;              /* Complementa el estado de la salida FL1
                               automaticamente */
    jump start;              /* Salta a la primera linea del programa */

```

Al código fuente mostrado anteriormente se lo puede dividir en cuatro partes, la primera es la inclusión de una librería específica para la arquitectura del microcontrolador DSP 2181:

```
#include <def2181.h>;
```

La segunda comienza con una definición de una sección de memoria de programa exclusiva para el manejo de interrupciones:

```
.section/pm interrupciones;
```

Lo que sigue a continuación es el llamado vector de interrupciones, nótese que para cada interrupción existen cuatro instrucciones *rti*, estas pueden ser reemplazadas por una subrutina de servicio a la correspondiente interrupción de hasta cuatro instrucciones reemplazando a los *rti*. Esto último puede ser ejemplificado para el *RESET* en donde se ha reemplazado el primer *rti* por:

```
jump start;
```

Que realiza un salto para empezar la ejecución del programa principal.

La tercera parte comienza con una definición de sección de memoria datos, donde se pueden incluir tablas, constantes y declaración de variables, en este caso se ha realizado esto último.

```
.section/dm variables;
```

La última parte comienza con una definición de sección de memoria de programa donde especifica el código fuente de la aplicación:

```
.section/pm programa;
```

Como puede verse claramente, la programación de los microcontroladores DSP ADI 21xx se la realiza tomando en cuenta las características de la arquitectura Harvard, así, en general, se deben fijar secciones de memoria de datos exclusivamente para almacenar tablas, constantes y declarar

variables, mientras que se define una segunda sección pero esta vez correspondiente a memoria de programa dentro de la cual se realizará la redacción para el código que implementará la funcionalidad prevista por el programador.

- Se crea un nuevo archivo, como se mencionó en pasos anteriores, con la diferencia que este deberá tener una extensión *.ldf*
- Una vez realizado el paso anterior se debe agregar el archivo fuente creado a la carpeta llamada *Linker Files* en la ventana de proyecto, esto se logra dando un clic derecho en dicha carpeta y escogiendo la opción *Add File(s) to Folder* en donde se visualizará un cuadro de diálogo donde encontraremos el archivo con extensión *.ldf* guardado anteriormente.

Dicho archivo *ldf* debe contener el siguiente código fuente que será descrito posteriormente:

```

ARCHITECTURE(ADSP-2181)

SEARCH_DIR( $ADI_DSP\218x\lib )

$OBJECTS = $COMMAND_LINE_OBJECTS;

MEMORY
{
    seccion_de_interrupciones { TYPE(PM RAM) START(0x00000)
END(0x0002f) WIDTH(24) }
    seccion_de_programa { TYPE(PM RAM) START(0x00030) END(0x03e80)
WIDTH(24) }
    seccion_de_variables { TYPE(DM RAM) START(0x00000) END(0x03e80)
WIDTH(16) }
}

PROCESSOR p0
{
    LINK_AGAINST( $COMMAND_LINE_LINK_AGAINST)
    OUTPUT( $COMMAND_LINE_OUTPUT_FILE )

    SECTIONS
    {
        seccion_de_interrupciones
        {
            INPUT_SECTIONS( $OBJECTS(interrupciones) )
        } >seccion_de_interrupciones
    }
}

```

```

seccion_de_programa
{
    INPUT_SECTIONS( $OBJECTS(programa))
} >seccion_de_programa
seccion_de_variables
{
    INPUT_SECTIONS( $OBJECTS(variables))
} >seccion_de_variables

}}

```

El código mostrado anteriormente puede ser dividido en tres partes para su explicación:

La primera consiste en dos líneas que establecen la referencia hacia la arquitectura del microcontrolador DSP ADI 2181:

```

ARCHITECTURE(ADSP-2181)
SEARCH_DIR( $ADI_DSP\218x\lib )

```

La segunda sección establece la declaración de las particiones de memoria tanto de programa como de datos en que se divide la misma, por ejemplo este proceso es análogo a particionar un disco duro:

```

MEMORY
{
    seccion_de_interrupciones { TYPE(PM RAM) START(0x00000)
END(0x0002f) WIDTH(24) }
    seccion_de_programa { TYPE(PM RAM) START(0x00030) END(0x03e80)
WIDTH(24) }
    seccion_de_variables { TYPE(DM RAM) START(0x00000) END(0x03e80)
WIDTH(16) }
}

```

Así, la memoria de programa se divide en exactamente dos porciones, la primera con un ancho de 48 localidades (0x00000 – 0x0002f), la segunda con un ancho de 15952 localidades (0x00030 – 0x03e80) con lo cual se comprueba el tamaño total de memoria de programa que es de 16k x 24 bits. La memoria de datos no se fragmentó y posee íntegramente las 16k localidades de 16 bits. Cabe señalar que las particiones aquí mencionadas no

tienen nada que ver con las secciones descritas en el archivo fuente de extensión *.dsp*

La tercera parte del archivo de extensión *.ldf* es:

```

PROCESSOR p0
{
    LINK_AGAINST( $COMMAND_LINE_LINK_AGAINST)
    OUTPUT( $COMMAND_LINE_OUTPUT_FILE )

    SECTIONS
    {
        seccion_de_interrupciones
        {
            INPUT_SECTIONS( $OBJECTS(interrupciones) )
        } >seccion_de_interrupciones

        seccion_de_programa
        {
            INPUT_SECTIONS( $OBJECTS(programa))
        } >seccion_de_programa
        seccion_de_variables
        {
            INPUT_SECTIONS( $OBJECTS(variables))
        } >seccion_de_variables

    }}

```

Aquí se ingresan los objetos dentro de las particiones de memoria antes descritas, dichos objetos corresponden a las secciones declaradas en el archivo fuente de extensión *.dsp*, pudiendo dentro de cada partición de memoria tener más de una sección, es decir, si en el archivo *.dsp* se han declarado varias secciones, estas pueden contenerse dentro de una misma partición. El modelo de archivo fuente *.ldf* mostrado aquí no necesariamente es el que siempre se utilizará, solamente se lo ha usado de forma didáctica y para ser funcional con el programa demo de *VisualDSP++ 3.5*. A continuación se presentará un esquema de un archivo *.ldf* diferente, el cual corresponde a un archivo *.dsp* más elaborado en el que se ha tenido que fijar más secciones de memoria y más particiones de la misma:

```

ARCHITECTURE(ADSP-2181)

SEARCH_DIR( $ADI_DSP\218x\lib )

```

```

{
  seg_inttab { TYPE(PM RAM) START(0x00000) END(0x0002f) WIDTH(24) }
  seg_code   { TYPE(PM RAM) START(0x00030) END(0x02fff) WIDTH(24) }
  seg_pmda   { TYPE(PM RAM) START(0x03000) END(0x03fff) WIDTH(24) }
  seg_data1  { TYPE(DM RAM) START(0x00000) END(0x00fff) WIDTH(16) }
  seg_data2  { TYPE(DM RAM) START(0x01000) END(0x01fff) WIDTH(16) }
}

PROCESSOR p0
{
  LINK_AGAINST( $COMMAND_LINE_LINK_AGAINST)
  OUTPUT( $COMMAND_LINE_OUTPUT_FILE )

  SECTIONS
  {
    seg_inttab
    {
      INPUT_SECTIONS( $OBJECTS(inicio) )
    } >seg_inttab

    seg_code
    {
      INPUT_SECTIONS( $OBJECTS(programa) $OBJECTS(programa1) )
    } >seg_code

    seg_data1
    {
      INPUT_SECTIONS( $OBJECTS(buffers) $OBJECTS(buffers1) )
    } >seg_data1

    seg_data2
    {
      INPUT_SECTIONS( $OBJECTS(data2) )
    } >seg_data2

    seg_pmda
    {
      INPUT_SECTIONS( $OBJECTS(datospm) )
    } >seg_pmda

  }}

```

En este ejemplo se ha particionado la memoria de programa tres veces para albergar al vector de interrupciones, al código que establece la funcionalidad de la aplicación y a datos que requieren una mayor precisión de la que se puede tener en la memoria de datos (coeficientes de filtros por ejemplo) respectivamente. La memoria de datos se la ha particionado en dos.

Adicionalmente se puede ver que en cada partición se ha albergado más de un objeto, correspondiendo a más de una sección de memoria en el archivo *.dsp*.

- Una vez que se han realizado todos los anteriores pasos se puede ya compilar el proyecto presionando F7 o haciendo *click* en el menú *Project* en la opción *Build Project*. Se observará entonces la ventana de salida, aquí se presentarán los resultados de compilación pudiendo mostrarse errores, advertencias con sus detalles o un mensaje de éxito.
- El proceso de compilado genera en la carpeta del proyecto creado una carpeta denominada *Debug* dentro de la cual existe un archivo de extensión *.dxe*, el cual hay que transformar a extensión *.exe* para poder cargarlo en la memoria del microcontrolador DSP a través del programa Host del Ez Kit Lite descrito en el numeral siguiente, no se genera el archivo *.exe* ya que esto se lo hacía en versiones anteriores, típicamente de DOS.

El proceso de generación del archivo *.exe* y su carga en la memoria del DSP se lo realizará en el siguiente numeral.

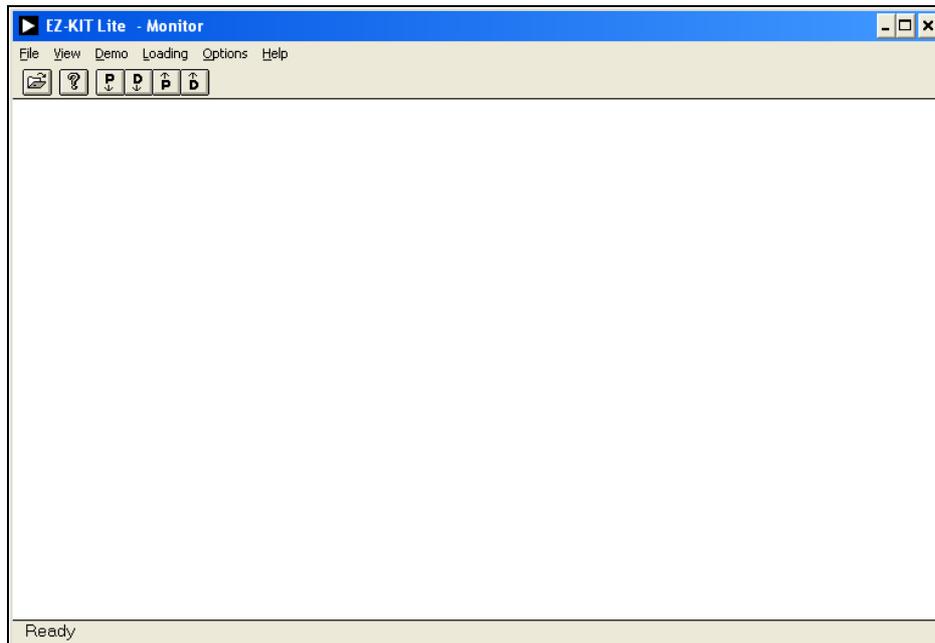
#### **1.4.4. PROGRAMA HOST EZ KIT LITE**

Este programa en su función principal permite descargar el archivo de extensión *.exe* directamente a la memoria interna del ADSP 2181, para ser ejecutado de inmediato como una subrutina del programa monitor, el cual reside de forma permanente en la memoria EPROM que está incluida en el kit de desarrollo<sup>27</sup>.

El entorno de dicho programa es el que se muestra:

---

<sup>27</sup> El programa monitor se carga inmediatamente a la memoria interna del ADSP 2181 luego de encendida la tarjeta



**Figura 1. 36:** Entorno del programa Host del EZ Kit

Para cargar el programa simplemente se escoge el ícono de abrir y se selecciona el archivo de extensión `.exe` desde su ubicación actual, asegurándonos además de que la conexión serial entre la PC y la tarjeta este libre. Para ello es necesario convertir el nuevo formato `.dxe` al viejo formato `.exe` a través de una herramienta provista por el mismo *Visual DSP++* en el directorio de instalación del mismo, que por defecto es:

*C:\Archivos de programa\Analog Devices\VisualDSP 3.5 16-Bit*

Se encuentra aquí el siguiente archivo ejecutable: *elf2aexe*, mediante el cual se realizará la transformación, entonces se procede de la siguiente forma:

Obviamente Windows no asociará la extensión `.dxe` a ningún programa, por lo cual usamos la opción *Abrir Con...* del menú desplegado al hacer clic derecho sobre el archivo `.dxe`, se busca el archivo ejecutable *elf2aexe* en la ruta especificada y se abre el `.dxe` con este, automáticamente se generan dos archivos uno de extensión `.sym` y otro de extensión `.exe` que es el que nos interesa.

Se debe asegurar que cuando se seleccione el programa *elf2a.exe* con la opción *Abrir Con...* se marque la casilla *Utilizar siempre el programa seleccionado para abrir este tipo de archivos*, con lo cual cada vez que se abra el archivo *.dxe* se generará automáticamente el archivo *.exe*.

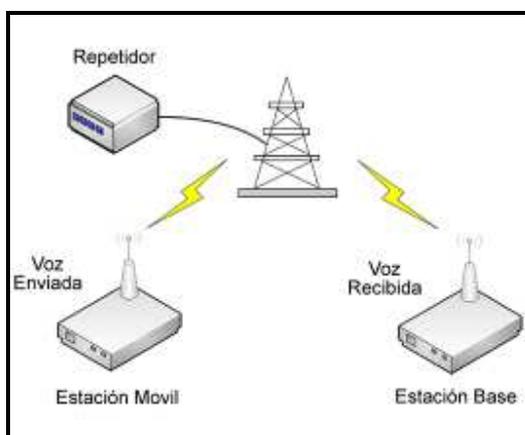
Para concluir con la realización del programa “*Hello World*” de *VisualDSP++ 3.5*, una vez cargado el archivo *.exe* se podrá observar el destello del led correspondiente a la bandera de salida *FL1*.

## 1.5. TRANSMISIÓN DE VOZ MEDIANTE RADIOS VHF

### 1.5.1. DESCRIPCIÓN BÁSICA

Las tecnologías de radio VHF, se han utilizado ampliamente para las comunicaciones de voz semi – dúplex. Una ventaja de estas tecnologías es que funcionan sin línea de vista entre los equipos terminales lo que permite la comunicación en sitios en donde otro método de comunicación sería difícil por las condiciones geográficas del entorno. La banda VHF, resulta ser entonces uno de los medios más fiables al momento de garantizar la comunicación en zonas de cobertura de corta y media distancia sin visibilidad directa, permitiendo tener una buena conexión de voz hasta dentro de un radio de 50 Km (según la zona geográfica).

Esta banda además presenta gran estabilidad, por cuanto no depende de las condiciones ambientales el momento de realizar los enlaces de comunicación.



**Figura 1. 37:** Sistema VHF orientado a voz

La implementación de un sistema VHF orientado a voz requiere mucho más de lo que aparenta la figura 1.35, se necesita de un estudio apropiado que tome en cuenta características como la zona geográfica y los alcances de la zona de cobertura, determinando así los parámetros más apropiados del sistema de comunicaciones y los equipos necesarios para tales efectos.

Adicionalmente se debe tomar en cuenta la parte legal, comúnmente los sistemas orientados a voz en VHF utilizan el modo semi – dúplex, por lo que habrá que concesionar un par de frecuencias portadoras (Transmisión y recepción) al organismo regulador competente.

Por lo general el ancho de banda máximo ocupado es el permitido por los equipos que son diseñados para la aplicación de voz, en el caso de las radio bases Kenwood y Motorola se tienen dos valores posibles 25 y 12.5 KHz., el primero de los mencionados no es posible utilizarlo por prohibición del organismo regulador ya que estos anchos de banda no son utilizables por razones de saturación en la *banda del servicio fijo – móvil terrestre*<sup>28</sup>.

## **1.5.2. MODOS DE OPERACIÓN**

La operación entre equipos de comunicación puede realizarse de dos maneras dependiendo del número de frecuencias usadas para transmisión y recepción de la información de voz.

### **1.5.2.1. Modo Simplex**

Por lo general este modo es de diagnóstico de los equipos y de corto alcance, en este se usa la misma frecuencia para transmitir como para recibir, de modo que el canal solamente puede ser usado por una de las dos partes, los equipos pueden

---

<sup>28</sup> Sección del espectro radioeléctrico destinado a los servicios de comunicación que operan con equipos de radio terrestres

comunicarse entre sí sin que sea necesario conectarles antenas externas dada las cortas distancias entre las mismas.

### **1.5.2.2. Modo Semi – dúplex**

Este modo de operación es el comúnmente usado en sistemas de gran cobertura en el que se hace indispensable el uso de un repetidor de señal, este modo emplea dos frecuencias portadoras, una para transmisión y otra para recepción, los equipos móviles y de central deben ser configurados con estas frecuencias, mientras que el repetidor, estratégicamente ubicado, debe ser configurado con las frecuencias cambiadas; los sistemas LAV por radiofrecuencia utilizan este esquema semi – dúplex.

### **1.5.3. DESCRIPCIÓN DEL EQUIPO EMPLEADO EN COMUNICACIONES VÍA RADIO**

En las comunicaciones vía radio existen radio bases, radios portátiles y repetidores, cada uno tiene características propias de la función que desempeñan.

Las radio bases son equipos diseñados para transmitir a una mayor potencia, por lo que su alcance será mayor, por tanto serán utilizadas en aplicaciones en las cuales el área de movilidad es considerablemente grande, por ejemplo una ciudad en un sistema de radio taxis.



**Figura 1. 38:** Radio base Kenwood

Las radios portátiles poseen una menor capacidad de transmisión en cuanto a alcance, por tanto serán usadas cuando el área sea considerada pequeña, por ejemplo en una construcción civil.



**Figura 1. 39:** Radio Portátil Kenwood

El repetidor es usado para retransmitir la señal enviada por uno de los equipos anteriores con el fin de lograr que el área de cobertura se expanda hasta los límites deseados.



**Figura 1. 40:** Repetidor Kenwood

#### **1.5.3.1. Modo de manipulación**

Previa a su operación los equipos deben estar dispuestos adecuadamente, para esto las frecuencias tanto de transmisión como de recepción deben estar configuradas en los equipos de manera correcta, otras opciones adicionales de programación de los equipos deben ser las mismas entre los miembros de un grupo de comunicación, el largo de las antenas debe corresponder a las frecuencias a las que se operan.

Para realizar una comunicación de voz simplemente se toma el micrófono y se presiona el botón *Push To Talk* mientras se habla, el otro lado escucha el mensaje y por tanto no podrá realizar ninguna transmisión de voz ya que el canal se encuentra ocupado, cuando se desocupe podrá realizarse una nueva transmisión de voz, todas las radios de un mismo grupo de interés deben estar sintonizadas en un mismo canal.

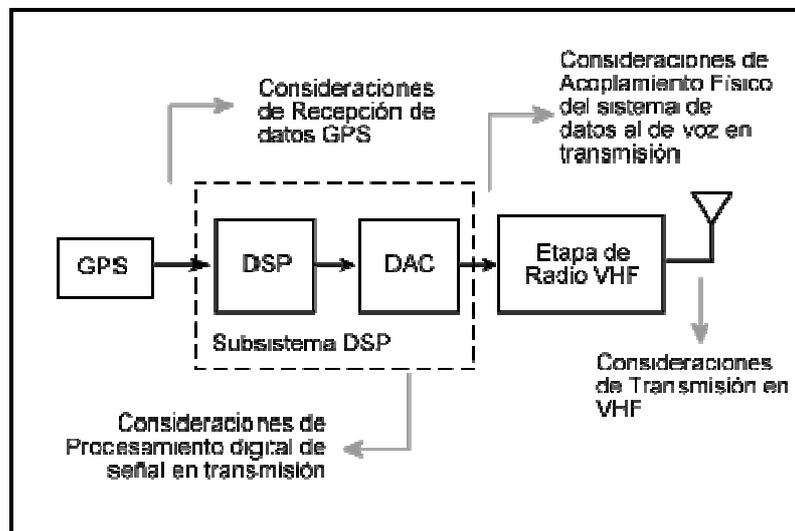
## CAPÍTULO 2

### CONSIDERACIONES DE DISEÑO

Este capítulo hace referencia a todos aquellos elementos que se necesitan tomar en cuenta para desarrollar los prototipos que forman parte del sistema básico de localización automática vehicular y que harán posible su implementación a través de radio en la banda VHF, como estos elementos establecen condiciones en dicho desarrollo, así como la estrategia elaborada en pro del cumplimiento de las expectativas del presente proyecto de titulación en función de su objetivo general.

Se realizan entonces análisis diferentes para los sistemas transmisor y receptor, señalando las consideraciones necesarias a tomar en cuenta en cada uno, con el objetivo principal de realizar un análisis lo más guiado posible.

#### 2.1. CONSIDERACIONES EN EL TRANSMISOR.



**Figura 2. 1:** Esquema básico estructural y consideraciones en sistema de transmisión

- **Consideraciones de recepción de datos GPS:** Describe las principales características de la recepción GPS que hay que tomar en cuenta para poder capturar y aprovechar los datos de navegación provistos y que son necesarios en la implementación del sistema de localización vehicular.
- **Consideraciones de transmisión VHF:** Establece las condiciones bajo las cuales se realizará la transmisión de la información en la banda VHF originando así las principales limitantes a las que debe ajustarse el sistema de procesamiento digital de señal, un ejemplo de estas limitantes es el ancho de banda disponible.
- **Consideraciones de Procesamiento Digital de Señal en transmisión:** Establece los lineamientos que deben seguirse en la implementación de los módulos a ejecutarse por el microcontrolador DSP, destacándose lo referido a la generación de onda portadora y su modulación adecuada con los datos de posicionamiento capturados, estableciendo de esta manera el transmisor de información de navegación.
- **Consideraciones de acoplamiento físico del sistema de datos al de voz:** Establece lo que hay que hacer para adaptar de manera física el sistema DSP representado en el Kit de desarrollo, a la radio que hará la transmisión en VHF, para establecer así el prototipo físico del transmisor.

### 2.1.1. RECEPCIÓN GPS

Receptor GPS es el dispositivo del cual se obtiene la información de navegación necesaria para la localización vehicular, el formato en que se entrega esta obedece a una estandarización realizada por un organismo norteamericano denominado NMEA<sup>29</sup> (*National Marine Electronics Association*), de tal manera que todos los receptores GPS alrededor del mundo esperan recibir datos en este formato, así que todo aquel sistema que procese esta información debe adaptarse al mismo, de ahí la importancia de los siguientes párrafos.

---

<sup>29</sup> [www.nmea.org](http://www.nmea.org)

La información de navegación consiste en un conjunto de líneas, denominadas tramas NMEA, existen diferentes tipos de estas, diferenciadas por el dispositivo hacia el cual se dirigen<sup>30</sup> y por el tipo de datos que llevan.

Cada nueva trama empieza con el ASCII \$ seguido por dos caracteres que identifican el tipo de dispositivo receptor, GP para los dispositivos GPS de uso general, seguidos a continuación por tres caracteres que identifican el tipo de información adjunta, que en la mayoría de los casos no pasa de los 80 caracteres de longitud; los diferentes campos de información contenida son separados por comas y finalmente la trama termina con lo que se conoce como “carrier return” y “line feed” dando paso a una nueva trama.

Como veremos más adelante todas las tramas incluyen un control de errores denominado checksum que comienza después del carácter “\*”, este control de errores puede o no ser verificado después de recibida la trama, para comprobar su integridad.

Existen tramas propietarias, es decir de fabricantes que han decidido incluir información personalizada, estas comienzan con la letra P en lugar de GP, ejemplos de dichos fabricantes son *Garmin*<sup>31</sup> y *Megallan*<sup>32</sup>, conocidos por sus dispositivos de navegación GPS.

#### **2.1.1.1. Aspectos de Interfaz del dispositivo GPS**

El estándar NMEA es compatible con la interfaz serial RS232 presente en cualquier computador, especifica una velocidad de transmisión de 4800 bps, con un modo de transmisión asíncrona, estableciendo ocho bits de datos, un bit de inicio y parada respectivamente, y sin paridad.

---

<sup>30</sup> En navegación existen otros tipos de dispositivos GPS que hacen uso de tramas NMEA tales como pilotos automáticos y localizadores de peces.

<sup>31</sup><http://www.garmin.com>

<sup>32</sup> <http://www.magellangps.com/>

Se debe notar que en la mayoría de dispositivos que incluyen un puerto serial la velocidad de transmisión configurada por defecto es de 9600 bps, pero los 4800 especificados por el estándar NMEA resultan ser más que suficientes; si consideramos que cada trama tiene una longitud de ochenta caracteres, tendríamos sesenta tramas por segundo que satisfacen cualquier necesidad de información actualizada en un instante dado.

La información dada y relacionada a los aspectos de interfaz del dispositivo GPS, resulta ser muy importante por cuanto el respectivo algoritmo que se implemente en el microcontrolador DSP, debe tomarla en cuenta para darle a este la capacidad de capturar los datos provistos de manera correcta y libre de errores.

#### 2.1.1.2. La Trama NMEA GPRMC

Con toda la información anteriormente proporcionada se puede dar un vistazo en un instante cualquiera a los datos entregados por un típico dispositivo GPS, el que se encuentra conectado a un computador común que posea un software terminal configurado a la velocidad de transmisión NMEA:

<b>Tramas GPS Recibidas</b>
<i>\$GPRMC,175707,A,0012.1195,S,07829.5335,W,000.0,032.2,140509,,,A*7D\r</i>
<i>\$GPGGA,175708,0012.1197,S,07829.5336,W,1,05,04.2,02810.2,M,017.9,M,,*4E\r</i>
<i>\$GPGSA,A,3,12,14,18,22,30,,,,,,,,,05.2,04.2,03.1*0D\r</i>
<i>\$GPGSV,3,1,12,01,41,084,00,06,06,245,00,12,18,060,46,14,21,341,45*7C\r</i>
<i>\$GPGSV,3,2,12,16,04,209,00,18,51,055,48,21,45,169,29,22,42,006,46*77\r</i>
<i>\$GPGSV,3,3,12,24,10,150,00,29,13,143,00,30,45,086,36,31,55,260,00*79\r</i>

**Tabla 2. 1:** Tramas GPS Recibidas

Todo el conjunto de tramas indicadas se generan cada segundo, de estas la denominada *GPRMC* resulta ser la más conveniente para los propósitos de un sistema LAV, por la información contenida en ella, que, como lo especifican los tres últimos caracteres de su nombre (RMC), es la mínima requerida para ayuda en la navegación en un sistema de posicionamiento global.

#### *2.1.1.2.1. Información proporcionada*

La trama NMEA GPRMC descrita campo a campo ofrece la siguiente información:

\$GPRMC,123519,A,4807.038,N,01131.000,E,022.4,084.4,230309,003.1,W\*6A

Dónde:

- 123519 Hora en Greenwich (12 Horas, 35 minutos y 19 segundos)
- A Estado del GPS, A = conectado a red posibilitando datos Actualizados, V = desconectado de la red no posibilitando actualizar datos de navegación.
- 4807.038,N Es la latitud, 48 grados 07.038 minutos Norte
- 01131.00,E Es la longitud, 11 grados 31.00 minutos Este
- 022.4 Velocidad sobre la tierra en Nudos<sup>33</sup>
- 084.4 Es el ángulo, en grados, respecto del Norte
- 230309 Representa la fecha actual en formato día, mes y año
- 003.1, W Variación Magnética<sup>34</sup>.
- \* 6A Checksum, siempre comienza con "\*" y se calcula entre, pero Sin incluir, los caracteres "\$" y "\*".

<sup>33</sup> Un nudo equivale a aproximadamente 1,852 Kilómetros por hora.

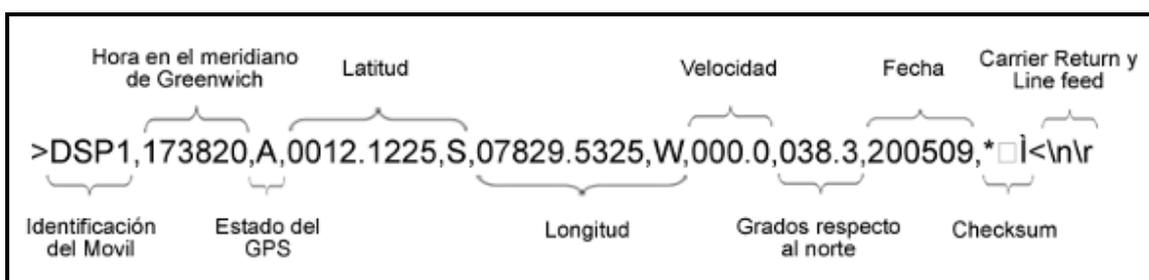
<sup>34</sup> Es el ángulo comprendido entre el norte geográfico y el norte magnético (norte indicado en una brújula)

Como puede visualizarse en la lista anterior, la información provista por esta trama NMEA resulta ser la adecuada para los propósitos de localización vehicular, ofreciendo además información de navegación complementaria que puede ser usada para implementar un sistema de información vehicular completo.

Entonces, es papel del algoritmo que se implemente en el micro procesador DSP, separar y almacenar la trama GPRMC del resto que se envían periódicamente, para aprovechar cada uno de sus campos, especialmente el referido a ubicación geográfica.

### 2.1.1.3. Trama de información a transmitir en el sistema básico LAV

Como puede verificarse en el numeral 2.1.1.2.<sup>35</sup> la trama NMEA GPRMC posee sesenta y ocho caracteres, algunos de estos no son útiles, es conveniente entonces modificar su formato para acoplar a los requerimientos de un sistema LAV propio y poder transmitir la nueva trama, así el conjunto de ASCIIs “GPRMC” se los puede reemplazar por el nombre del dispositivo móvil que envía la trama, por ejemplo DSP1, el campo correspondiente a la variación magnética por lo general se encuentra vacío, lo cual puede ser verificado en la tabla 2.1, pudiendo ser eliminado por completo, de esta manera las tramas a enviar lucen como la siguiente:



**Figura 2. 2:** Trama de posicionamiento a transmitir en el sistema LAV

Sesenta y cinco caracteres en total.

<sup>35</sup> Página 73 del presente proyecto de titulación

Nótese que en la trama de posicionamiento a transmitir se incluye un checksum, lógicamente este no es el mismo al original, pues al eliminar campos de la trama inicial este se alteró, entonces el que se indica es uno nuevo que se debe calcular.

### **2.1.2. TRANSMISIÓN EN VHF**

Las características para transmisión en la banda VHF, y en general para hacerlo en cualquier franja del espectro radioeléctrico, resultan ser determinantes en el diseño de cualquier enlace de comunicaciones inalámbrico, no solo por las diferentes particularidades de propagación de cada banda de frecuencias, sino también por las características de los equipos disponibles e inclusive por barreras de tipo regulatorio.

Además, las exigencias de la aplicación que se desea implementar también determinan los recursos mínimos necesarios, se debe realizar entonces un análisis de lo que se tiene y lo que se quiere, concluyendo lo que se puede conseguir y con qué metodología.

En el caso particular de transmisión de datos, el plan nacional de frecuencias establece rangos apropiados para este fin<sup>36</sup>, sin embargo, hay situaciones en las cuales se desea hacer uso de una tecnología probada y de una infraestructura ya instalada, es este nuestro caso, en el cual se desea enviar y recibir información de localización vehicular sobre un sistema netamente de voz.

#### **2.1.2.1. Ancho de banda del canal disponible**

En nuestro país se comercializan principalmente dos marcas de radios para trabajar en el servicio fijo – móvil terrestre de voz, Kenwood y Motorola, ambas permiten trabajar con franjas de banda ancha o estrecha de 25 y 12.5 KHz respectivamente; como se mencionó en el numeral 1.5.1<sup>37</sup> en el Ecuador las normas regulatorias impiden hacer uso del ancho de banda de 25 KHz por razones de saturación.

---

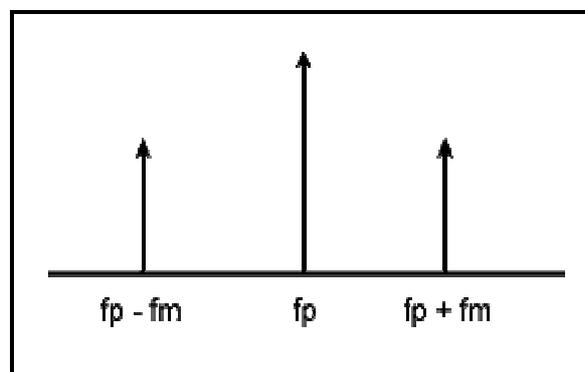
<sup>36</sup> CONATEL - SENATEL, Plan Nacional de Frecuencias, ECUADOR, septiembre, 2000, pp. 142 –EQA - 150

<sup>37</sup> Página 65 del presente proyecto de titulación

Se hace énfasis en esta situación por cuanto el ancho de banda del que se puede hacer uso resulta fundamental en las características de transmisión que se pueden obtener; siendo este un recurso limitado hay que buscar una metodología que permita satisfacer los requerimientos de la aplicación.

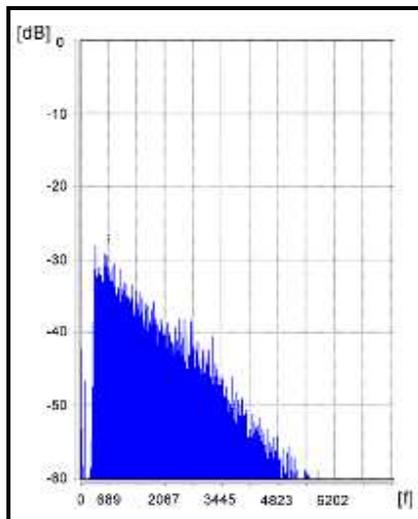
Como sabemos de la teoría de comunicaciones, a una creciente velocidad de señal el ancho de banda necesario será mayor, entonces la mencionada metodología consiste en buscar un esquema de modulación lo más eficiente posible, en términos de obtener un sistema lo suficientemente rápido y que sea fiable en el menor ancho de banda, dada una relación señal a ruido.

Si bien se menciona como el ancho de banda manejado por los equipos y permitido por el ente regulador los 12.5 KHz, estos no son aprovechados en su totalidad por cuanto los radios utilizan un esquema de modulación FM de tipo banda angosta llamado 11K0F3E.



**Figura 2. 3:** Espectro FM de banda angosta

Como puede apreciarse en la figura 2.3, este tipo de modulación FM de banda angosta replica la información ( $fm$ ) en ambos lados de la frecuencia portadora ( $fp$ ), de tal modo que los 12.5 KHz se convierten en 6.25 KHz aprovechables en banda base, esta última suposición basada en la existencia de un filtro ideal.



**Figura 2. 4:** Espectro en recepción en respuesta a una señal de ruido blanco

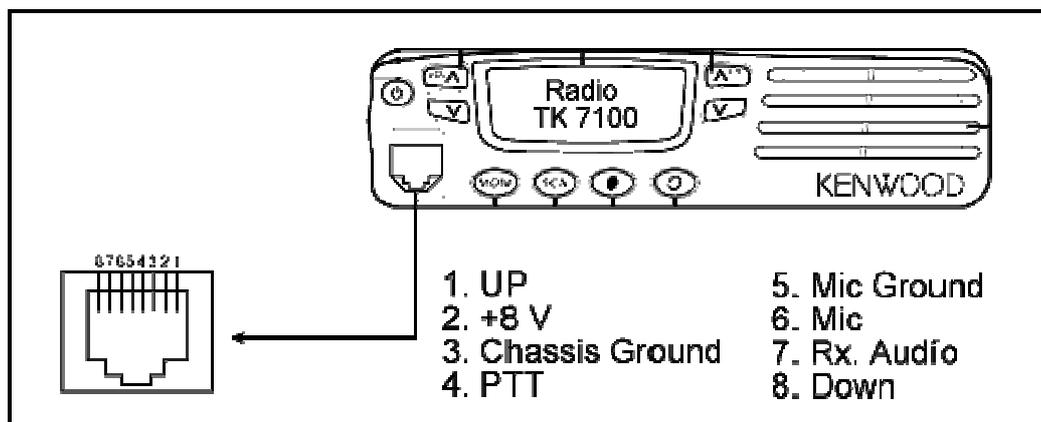
La figura 2.4 muestra el espectro de una señal correspondiente a ruido blanco que ha atravesado un canal de radio de 12.5 KHz de ancho de banda, se puede confirmar entonces la no existencia de un filtro ideal, por lo que los 6.25 KHz no son aprovechables en su totalidad sino una fracción de los mismos, por lo que se considera aprovechable hasta un máximo de 4 KHz.

#### **2.1.2.2. Frecuencias portadoras disponibles en modo semi – dúplex**

El organismo regulador competente es el encargado de concesionar un par de frecuencias portadoras para transmisión y recepción respectivamente, esto es importante ya que las frecuencias así obtenidas deben estar dentro de la banda aceptada por los equipos de radio, que típicamente se los puede clasificar como de rango alto o bajo, dependiendo de la franja radioeléctrica en la cual pueden operar.

#### **2.1.3. ACOPLAMIENTO FÍSICO DE VOZ Y DATOS EN TRANSMISIÓN**

Para la implementación física del prototipo se debe acoplar la tarjeta de desarrollo Ez Kit Lite a la radio base transmisora, de tal modo que esta comande el envío de la información de navegación, para tal fin se debe utilizar las entradas provistas por la radio base a usar.



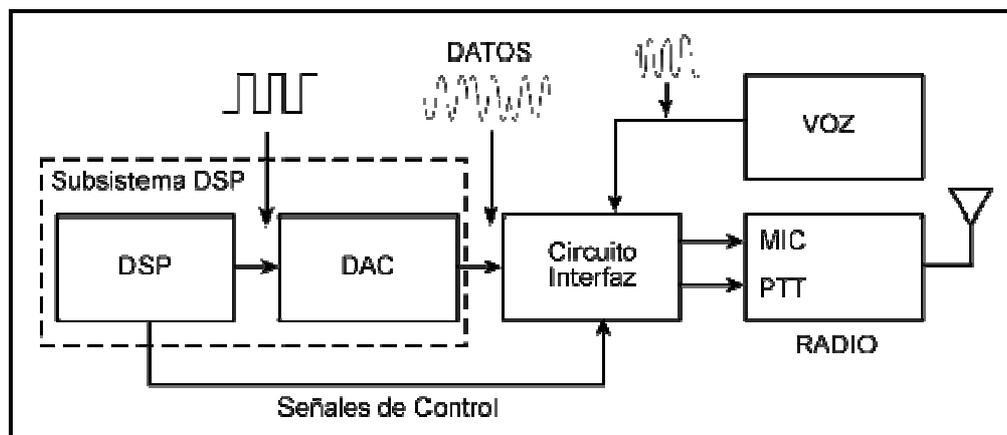
**Figura 2. 5:** Entradas de señal de la radio base TK – 7100

En la figura 2.5 puede visualizarse el jack RJ – 45 que da acceso a los pines de entrada de la radio y su respectiva función, de todos estos los esenciales para el acoplamiento son el número 4 y el 6 que corresponden a las señales de *Push To Talk* y *Mic* respectivamente, la primera de las señales mencionadas maneja la apertura del canal para que el usuario pueda hacer su transmisión de voz a través del micrófono cuya señal se lleva a la entrada *Mic*.

La transmisión de información de navegación, usando la señal *Mic*, debe ser automática de modo que la apertura del canal a través de la señal *Push To Talk* ya no es manual como en la transmisión por voz, sino debe ser manejada por algún algoritmo implementado en el microcontrolador DSP que establezca el momento oportuno de la transmisión de información.

Tal como se plantea, los pines 4, 6 y sus respectivas referencias deben ser conectados al Kit de desarrollo para su correspondiente manejo, esto elimina por completo la posibilidad de transmisión por voz, se debe entonces diseñar un circuito que sirva de interfaz entre la tarjeta de desarrollo y la radio base, de modo que se pueda manejar tanto transmisión de voz como de datos, sin mutua interferencia el instante de ocupar el canal inalámbrico de comunicaciones.

A continuación se presenta un esquema ilustrativo de esta consideración:



**Figura 2. 6:** Esquema de Acoplamiento físico en transmisor

Como puede visualizarse en la figura anterior el circuito interfaz deberá tener los elementos adecuados para poder conmutar entre voz y datos, dicha conmutación es manejada por el microcontrolador DSP a través de señales de control, así como también la apertura automática del canal para poder transmitir la información de navegación.

#### **2.1.4. PROCESAMIENTO DIGITAL DE SEÑAL EN TRANSMISIÓN**

Con toda la información necesaria y suficiente de los elementos a considerar, se procede a organizar el esquema de los módulos a ejecutarse mediante el microcontrolador DSP y que conformarán el transmisor de información de navegación, parte fundamental del sistema LAV, por cuanto permitirá montar una estructura dedicada de datos sobre otra pensada inicialmente para soportar solamente voz.

El procesamiento digital de señal en el transmisor se lo ha dividido en módulos de tal manera que cada uno se encargue de una tarea específica, facilitando de esta manera el desarrollo, depuración y posterior entendimiento del sistema en discusión. A continuación se presenta un esquema ilustrativo de lo que deberá realizarse en el microcontrolador DSP y además como este interactúa con el exterior obteniendo datos y generando señales de salida.

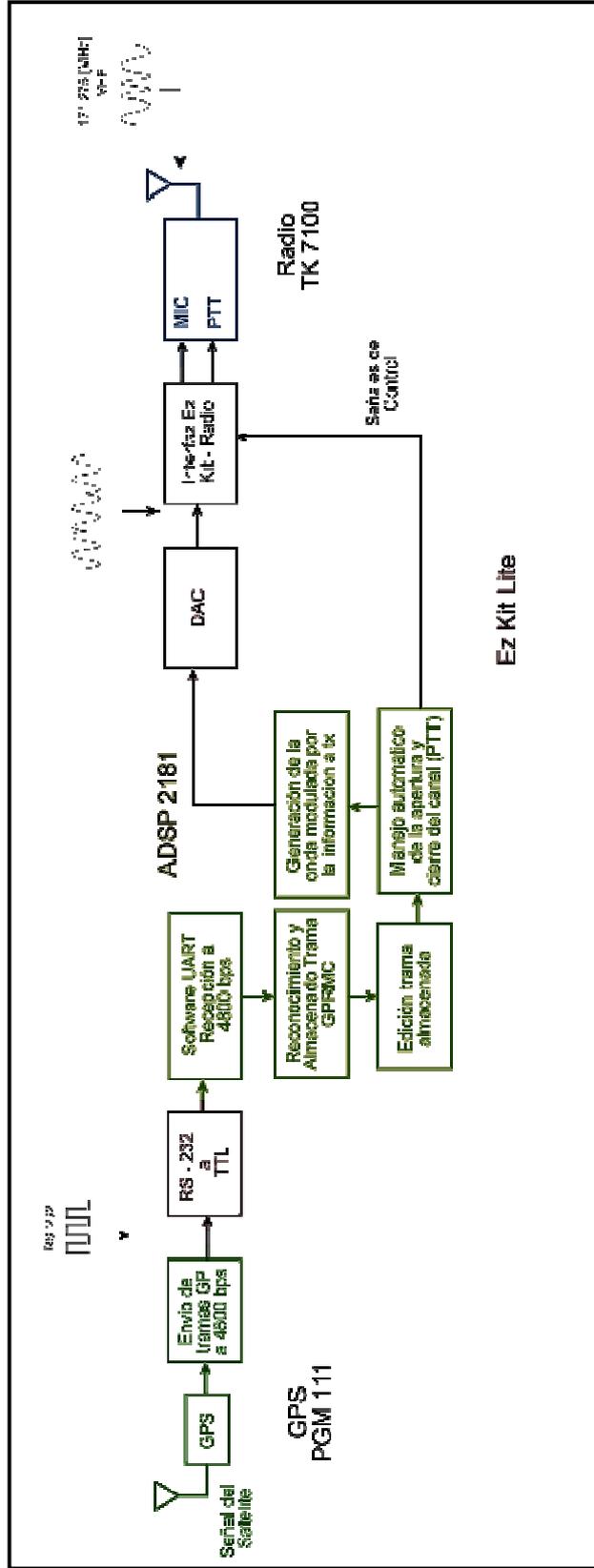


Figura 2. 7: Conformación del Transmisor

En la figura 2.7 se puede apreciar claramente los módulos a ejecutarse en el microcontrolador DSP, alguno generará señales de salida de control con el fin de comandar la transmisión de información de navegación, pero quizás el más importante constituye el bloque de modulación de información cuyas consideraciones y las de los demás bloques se detallan a continuación.

#### **2.1.4.1. Módulo de Generación de onda modulada**

Este constituye uno de los procesos más importantes que debe realizarse mediante el sistema DSP, por cuanto por definición, permite adaptar la información que se desea transmitir al canal de comunicaciones, en este caso el inalámbrico, de forma que el envío se lleve a cabo de la manera más eficiente posible.

El esquema de modulación a escoger debe considerar el ancho de banda utilizable del cual se dispone, en nuestro caso se llegó a deducir que es de 4 KHz en el numeral 2.1.2.1.<sup>38</sup>, ya que como se mencionó, a una creciente velocidad de modulación, mayor será el ancho de banda requerido, entonces hay una limitación en cuanto a la velocidad de señal que se puede obtener para considerar despreciable el número de errores en recepción, a este límite se lo conoce como capacidad del canal:

$$C = B * \log_2(1 + S/N) \quad (2.1)$$

Donde

C = Capacidad del canal en Bits por segundo

B = Ancho de banda del canal

S/N = Relación Señal a ruido (adimensional)

---

<sup>38</sup> Página 71 del presente proyecto de titulación

Para calcular la capacidad de canal se necesita entonces el parámetro señal a ruido, su valor depende de las características de los equipos transmisores de señal, así como de condiciones físicas, un valor típico es aproximadamente 48 dB<sup>39</sup>, con esta S/N se realiza el cálculo de la capacidad del canal para el caso de interés:

$$C = 4000 * \log_2(1 + 63.1 * 10^3)$$

$$C = 63.78 \text{ kbps}$$

Si la tasa de información ( $V_{tx}$ ) escogida, es igual o menor que  $C$  debe existir una técnica de modulación que permite una transmisión con un BER arbitrariamente pequeño, si  $V_{tx}$  es mayor que  $C$  entonces no es posible una transmisión libre de errores por más buena técnica que se use; si la  $V_{tx}$  escogida se aproxima a la capacidad del canal, la técnica de modulación que lo haga posible será más difícil de implementar, no existiendo una que haga posible utilizar un canal en toda su capacidad sin una recepción libre de errores.

Hemos determinado entonces un límite superior para la velocidad de señal, el cual como se ha visto no es alcanzable de manera exacta, para tratar de usar el canal en un buen porcentaje de su capacidad se requerirían de complejos esquemas de modulación multinivel.

El límite inferior está determinado por las condiciones de uso normal del canal de voz sobre el cual se enviarán los datos de navegación, cuan frecuente es la necesidad de estos y el número de móviles a monitorear, de modo que haya mínimas pérdidas de información de voz, no por interferencia de datos, sino por existir una sola fuente ocupando el canal en un instante dado.

Se tendrán en cuenta las siguientes consideraciones para obtener un límite inferior razonable:

---

<sup>39</sup> LEDGER, Dan y TOMARAKOS, John; Using The Low Cost High Performance ADSP – 21065L Digital Signal Processor For Digital Audio Applications; Primera Edición; Analog Devices; Norwood, USA, 1998; pp. 11

- 50 Móviles a monitorear.
- Base de tiempo de 60 segundos.
- 50% de actividad en voz y 50% de actividad para localización, tomando como referencia la base de tiempo.

Se realizan entonces los siguientes cálculos:

$$\textit{T tiempo disponible Datos} = 30 \textit{ seg}$$

El resultado siguiente da la duración de trama de localización por móvil:

$$\textit{T tiempo disponible para localizar un vehículo} = \frac{30 \textit{ seg}}{50 \textit{ móviles}} = 0.6 \frac{\textit{seg}}{\textit{móvil}}$$

$$\textit{C Caracteres por trama} = 65$$

$$\textit{t tiempo necesario por caracter} = \frac{0.6 \frac{\textit{seg}}{\textit{móvil}}}{65 \textit{ caracteres}} = 0.0092307692 \frac{\textit{seg}}{\textit{movil} * \textit{caracter}}$$

$$\textit{b bits por caracter} = 11 \frac{\textit{bits}}{\textit{caracter}}$$

$$\textit{t tiempo de bit} = \frac{0.0092307692 \frac{\textit{seg}}{\textit{movil} * \textit{caracter}}}{11 \frac{\textit{bits}}{\textit{caracter}}} = 0.0008391608392 \frac{\textit{seg}}{\textit{movil} * \textit{bits}}$$

$$\textit{V Velocidad de transmisión} = \frac{1}{\textit{t tiempo de bit}} = \frac{1}{0.000839160 \frac{\textit{seg}}{\textit{movil} * \textit{bits}}} = 1192 \frac{\textit{bps}}{\textit{móvil}}$$

Por tanto el límite inferior de velocidad de transmisión para las condiciones dadas anteriormente resulta ser de 1200 bps, el cual constituye el valor a implementar en el desarrollo de la presente aplicación.

El esquema de modulación debe considerar además las condiciones del canal de comunicaciones y como este afecta a una señal que transita por el mismo, típicamente se hace variar un parámetro (amplitud, fase o frecuencia) de una onda portadora senoidal en función de la información a transmitir, se debe escoger entonces el menos susceptible a errores debido a la influencia del canal de comunicaciones inalámbrico, ya que de esto depende en buena parte la robustez del sistema de comunicaciones y el rendimiento del sistema LAV completo.

Se ha visto entonces la necesidad de considerar en buen esquema de modulación que de un soporte fiable a la transmisión de datos, tomando en cuenta las limitaciones en cuanto a los recursos de los que se dispone.

Se vio también un límite superior para la velocidad de transmisión y se concluyó que se podría alcanzar un buen porcentaje del mismo si se usa un esquema de modulación multinivel, sin embargo el cálculo del límite inferior, tomado como característica de diseño, permite el uso de esquemas de dos niveles (binario), de esta manera se evaluará posibilidades dentro de estos esquemas, los cuales por sus propias características tendrán un límite superior de la velocidad de señal que puedan alcanzar dado el ancho de banda utilizable.

El tipo de modulación a emplear se encuentra entonces entre lo que se conoce como radio digital, definida como la transmisión de portadoras analógicas moduladas digitalmente, este término no se debe confundir con transmisión digital, ya que este último consiste en el envío de pulsos digitales, sin la intervención de un proceso de modulación.

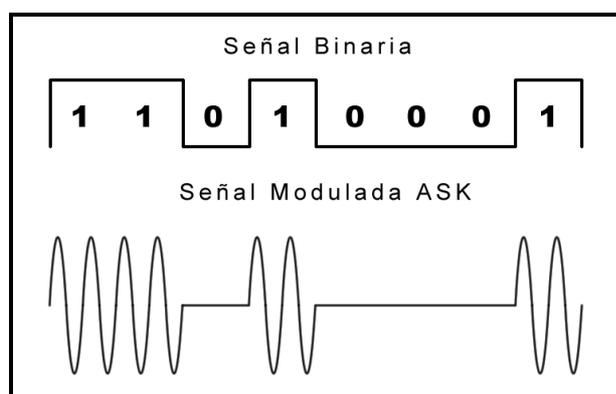
La radio digital agrupa a tres principales clases de modulación: ASK, FSK y PSK, diferenciándose una de la otra en el parámetro tomado como variable dependiente de la modulante.

Los siguientes numerales están orientados a realizar una descripción objetiva de cada tipo, para posteriormente con los suficientes elementos de juicio, determinar la idoneidad de cada una de estas modulaciones para implementar su respectiva forma

de onda en el microcontrolador DSP, que de esta forma hará posible el envío de información de navegación sobre un canal pensado exclusivamente para voz. Así en el presente capítulo se hará referencia a consideraciones de ancho de banda y su relación con la máxima velocidad de transmisión que se puede lograr con el correspondiente esquema de modulación, claro está tomando en consideración el ancho de banda limitado del cual se dispone que se determinó como de 4 KHz, todo lo cual permitirá concluir en primera instancia el esquema adecuado.

#### 2.1.4.1.1. Modulación ASK

ASK (*Amplitude Shift Keying*) es la más sencilla de las modulaciones en radio digital.



**Figura 2. 8:** Modulación ASK

En la figura 2.8 pueden apreciarse la cadena de bits que se desean transmitir y el resultado de emplear este esquema de modulación, donde puede apreciarse que para un cero lógico se suprime por completo la amplitud de la portadora y para un uno lógico se tiene la amplitud máxima de la misma, por ello se le denomina a menudo OOK (On – Off Keying) de portadora encendida o apagada.

El ancho de banda necesario para ASK se relaciona directamente con la velocidad de señal y esta a su vez con el ancho de banda de la señal digital a través del criterio de Nyquist para señales de banda base:

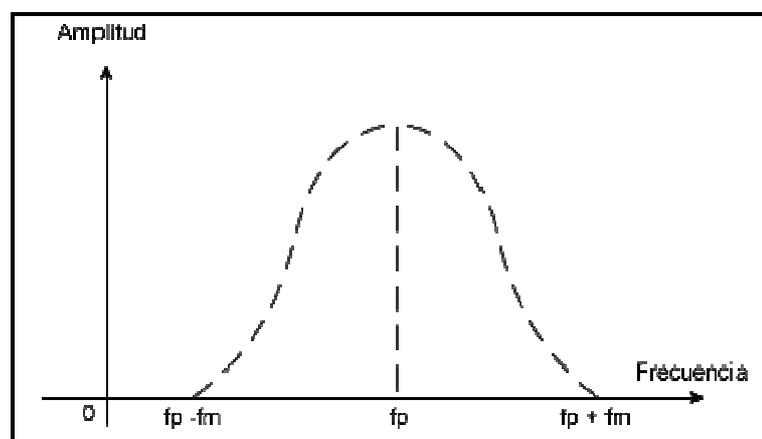
$$V_s = 2B \quad (2.2)$$

Dónde:

$V_s$  = Velocidad de señal

$B$  = Ancho de banda limitado de la señal digital

Espectralmente la modulación ASK produce un traslado de frecuencias de banda base hacia la frecuencia de portadora, de manera similar a AM:

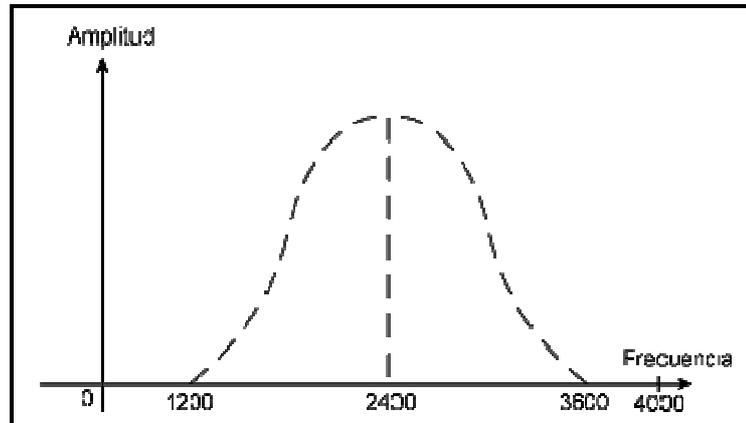


**Figura 2. 9:** Espectro ASK

En la figura 2.9  $f_p$  representa la frecuencia de portadora y  $f_m$  la máxima frecuencia observable de la señal digital limitada en banda, de modo que la máxima velocidad de señal que se puede obtener es  $2 f_m$  y es también el ancho de banda ocupado por ASK.

Dado el ancho de banda utilizable de 4 KHz. la máxima velocidad de señal que se puede alcanzar con este esquema, con una portadora de frecuencia de 2400 Hz<sup>40</sup> es de 2400 baudios, tal y como se puede concluir de la siguiente ilustración:

<sup>40</sup> Cantidad tomada por ser la más cercana al centro del ancho de banda disponible y que es múltiplo de la frecuencia de muestreo que se utilizará (48000Hz), esto último considerado para facilidad de generación de portadora en el microcontrolador DSP

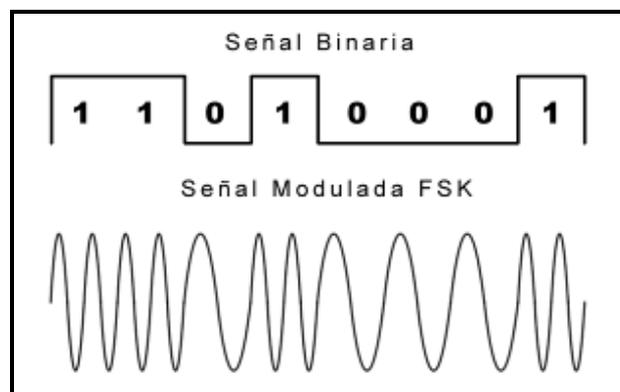


**Figura 2. 10:** Capacidad ASK

Como puede verse en la figura 2.10, si la velocidad de señal sube más se acercaría al límite en ancho de banda del canal donde las frecuencias son más atenuadas.

#### 2.1.4.1.2. *Modulación FSK*

En FSK (Frequency Shift Keying) el parámetro que varía en función de la información digital es la frecuencia:



**Figura 2. 11:** Modulación FSK

Cada uno de los estados lógicos de la señal digital son representados por distintas frecuencias de portadora, donde típicamente el valor más bajo de frecuencia corresponde a un cero lógico y el alto a un uno lógico, comúnmente se les llama frecuencia de espacio y frecuencia de marca respectivamente.

Se puede demostrar que el ancho de banda necesario para una señal FSK viene dado por la siguiente expresión:

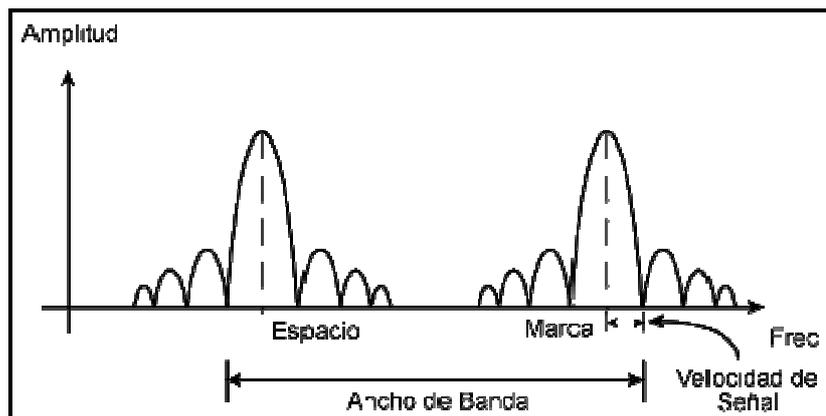
$$B = 2(\Delta f + f_b) \quad (2.3)$$

Donde:

$B$  = Ancho de Banda de la señal FSK

$f_b$  = velocidad de transmisión

$$\Delta f = |\text{frecuencia de marca} - \text{frecuencia de espacio}| / 2$$

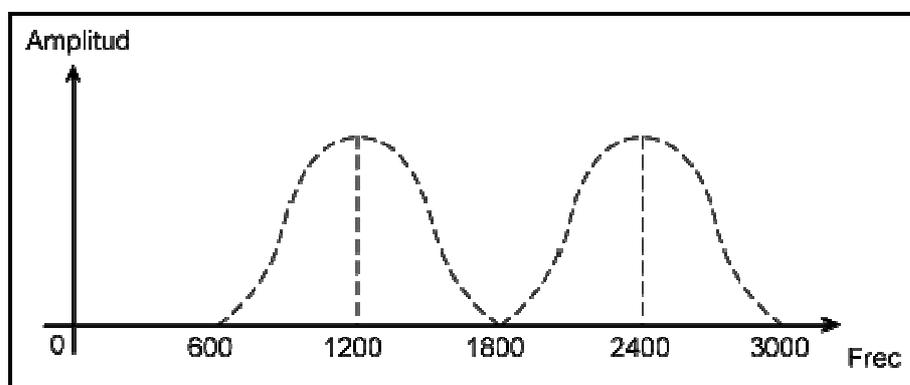


**Figura 2. 12:** Espectro FSK

A menudo, en el contexto de modulación, se suele confundir los términos *velocidad de transmisión* y *velocidad de señal*, el primero se refiere a la velocidad con la que los bits alimentan al modulador, el segundo se refiere a la velocidad con la que los símbolos salen del mismo, entendiéndose a la modulación desde este punto de vista, como el proceso que transforma bits en símbolos. Para el caso de señales digitales multinivel cada símbolo a la salida del demodulador contendrá más de un bit, para señales digitales binarias cada símbolo representa un bit, en este último caso los valores numéricos de velocidad de transmisión y velocidad de señal coinciden, no así sus unidades, ya que la primera está en bps y la segunda en baudios.

El ancho de banda de la señal FSK depende entonces de las frecuencias que se tomen como de marca y de espacio, y claro está, también de la velocidad de transmisión.

Dado el ancho de banda utilizable de 4 KHz la máxima velocidad de señal que se puede alcanzar con este esquema, con una frecuencia de marca de 2400 Hz y una frecuencia de espacio de 1200 Hz, es de 600 baudios, tal y como se puede concluir de la siguiente ilustración:

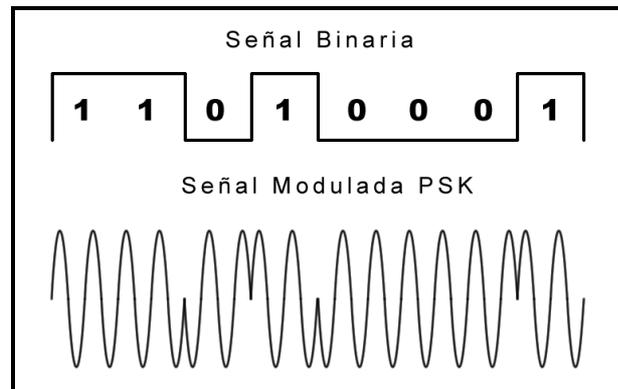


**Figura 2. 13:** Capacidad FSK

Como puede verse en la figura anterior, si la velocidad de señal sube más habrá un traslape de espectros, por lo que se producirán errores en la demodulación.

#### **2.1.4.1.3. Modulación BPSK**

En BPSK (Binary Phase Shift Keying) el parámetro que varía en función de la modulante digital es la fase de la portadora senoidal.



**Figura 2. 14:** Modulación BPSK

La fase de una señal senoidal, puede entenderse si se quiere, como el ángulo desde donde comienza la onda, típicamente para representar un cero lógico se usa una onda senoidal que comienza en 180 grados, para representar el uno lógico una que comience en cero grados, claramente hay un desfase de 180 grados entre un estado y otro, lo cual es corroborado por la figura 2.14.

Se puede demostrar que el ancho de banda de una señal BPSK viene dado por la sencilla expresión:

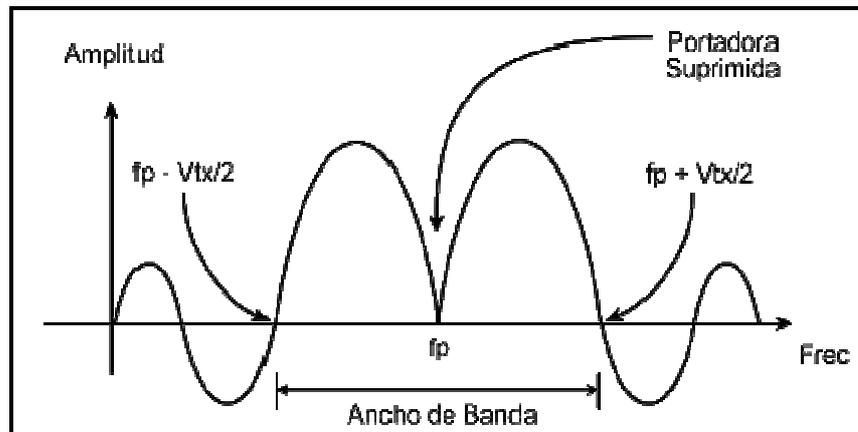
$$B = V_{tx} \quad (2.4)$$

Donde:

$B$  = Ancho de banda de la señal BPSK

$V_{tx}$  = Velocidad de transmisión

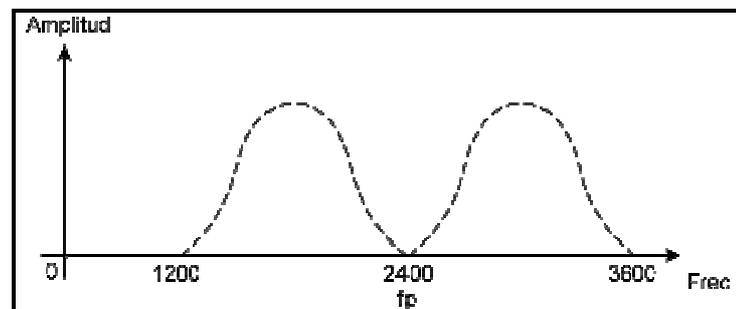
Entonces el ancho de banda absolutamente necesario para una señal BPSK es igual a la velocidad de transmisión escogida.



**Figura 2. 15:** Espectro BPSK

Como lo indica la figura 2.15 la portadora esta suprimida, por lo que el proceso de demodulación necesariamente debe ser coherente, es decir se tiene que reinsertar la portadora.

Dado el ancho de banda utilizable de 4 KHz la máxima velocidad de señal que se puede alcanzar con este esquema, con una portadora de frecuencia de 2400 Hz es de 2400 baudios, tal y como se puede concluir de la siguiente ilustración:



**Figura 2. 16:** Capacidad PSK

A una mayor velocidad de señal se acerca al límite del ancho de banda del canal donde hay una atenuación de frecuencias.

Se puede concluir en primera instancia que los esquemas que permite alcanzar el objetivo de los 1200 bps enunciado en el numeral 2.1.4.1.<sup>41</sup> son ASK y PSK, se tiene entonces un primer elemento de juicio para determinar el esquema de modulación que será implementado, restando por determinar en la parte práctica las facilidades de implementación y susceptibilidad al ruido en un canal de comunicaciones inalámbrico real, lo cual se realizará en un capítulo posterior.

#### **2.1.4.2. Módulo de recepción y almacenamiento de carácter desde el GPS**

Este módulo es el que se debe relacionar directamente con la información provista por el dispositivo GPS, para lo cual se debe considerar lo señalado en el numeral 2.1.1.1<sup>42</sup> en cuanto a la velocidad de transmisión NMEA, para ello se debe implementar un software UART (*Universal Asynchronous Receiver - Transmitter*) que permita capturar los datos y almacenarlos para inmediatamente analizarlos.

Para este proceso de análisis se debe tomar como unidad el carácter, es decir cuando se ha recibido completamente uno de ellos se procede a realizar alguna operación con él, la misma que es efectuada por el siguiente módulo de procesamiento. Cabe señalar que este módulo solo debe guardar el carácter actual, la siguiente etapa se debe encargar de la decisión de almacenarlo o desecharlo de forma definitiva.

#### **2.1.4.3. Módulo de detección y almacenamiento de la trama NMEA GPRMC**

La operación que debe realizarse con el carácter capturado por el módulo de procesamiento anterior es precisamente verificar si se trata de la trama GPRMC, y es esta la razón por la cual se toma como unidad de procesamiento al carácter, porque nos permite llevar a cabo esta tarea realizando una simple comparación de los tres últimos caracteres que son los que determinan el tipo de trama captada por el dispositivo GPS.

---

<sup>41</sup> Página 82 del presente proyecto de titulación

<sup>42</sup> Página 72 del presente proyecto de titulación

De determinarse que los caracteres entrantes pertenecen a la trama GPRMC se debe iniciar un proceso de almacenamiento de los mismos en un buffer destinado para tal fin, caso contrario se desechan y continúa indefinidamente el proceso de búsqueda.

#### **2.1.4.4. Módulo de edición de la trama almacenada**

Esta etapa del procesamiento digital de señal en transmisión es la encargada de modificar la trama GPRMC originalmente almacenada para generar la trama definitiva a modular, esto según las especificaciones dadas en el numeral 2.1.1.3<sup>43</sup>, esto incluye la eliminación del campo GPRMC y su reemplazarlo por DSP1, eliminar el campo de variación magnética, generación de un nuevo checksum y finalmente incluir los caracteres de inicio y fin de trama (><).

#### **2.1.4.5. Módulo de evaluación de apertura automática del canal**

Según lo indicado en el numeral 1.5.3.1<sup>44</sup> en el sistema original, diseñado para voz, el usuario que desea comunicarse realiza un *Push To Talk*, mientras que la transmisión de información de navegación no debe depender de dicho usuario, esta etapa entonces debe manejar de manera automática el acceso al canal inalámbrico de comunicaciones a través de la señal de activación del *Push to talk*, que abre el canal de comunicaciones a la información de localización vehicular.

La automatización llevada a cabo por esta etapa debe relacionarse con diferentes situaciones que generan información y hacen necesario la apertura del canal de comunicaciones para transmitir los datos de navegación.

La trama NMEA GPRMC se genera una vez cada segundo, lógicamente su información no debe ser enviada con la misma frecuencia por cuanto acapararía el uso del canal, se debe enviar en cada evento en que se considere se genera

---

<sup>43</sup> Página 75 del presente proyecto de titulación

<sup>44</sup> Página 69 del presente proyecto de titulación

información, aprovechando para ello los distintos campos presentes en la trama GPRMC.

Existen tres casos principales que se consideran referenciales en la decisión de transmitir o no información de localización:

- El móvil se encuentra detenido.
- El móvil se encuentra en movimiento.
- El móvil se encuentra en estado de pánico.

De los tres casos mencionados, el primero es el que menos información genera, ya que el móvil se encuentra estático en el transcurso del tiempo, desde este punto de vista el intervalo entre transmisión de información sobre su localización sería más largo que los posteriores casos.

El segundo caso genera información de posición que se actualiza cada instante, la información sobre su estado se debe transmitir entonces con mayor frecuencia; tomando en cuenta parámetros como velocidad media y distancia recorrida en el cálculo adecuado del intervalo entre transmisiones de información de localización.

El último caso es el más crítico, pues el usuario activa el mismo en casos de extrema urgencia donde necesita de soporte inmediato, por ejemplo el robo del vehículo, en esta situación la información sobre la localización exacta del vehículo debe comunicarse entre intervalos cortos de tiempo, para tomar las acciones pertinentes.

Así mismo esta etapa debe considerar el hecho de que se está diseñando un sistema de información de localización sobre otro que fue ideado netamente para la transmisión de voz, es decir la transferencia de la misma es prioritaria a la relacionada con la ubicación del móvil, de tal manera que el sistema de voz permanezca inalterado y a la vez no se produzcan pérdidas de datos de localización.

Por ejemplo, se establece la necesidad de recibir información de localización vehicular cada minuto en un estado de permanente movimiento, en el momento de envío de datos no se debe interferir con la transmisión de voz, suponiendo que en dicho instante el usuario este haciendo uso normal de la radio base; en este ejemplo al tener prioridad voz, los datos de ubicación del móvil, para ese instante, se perderían, lo cual es inaceptable, por tanto esta etapa se debe encargar de detectar el instante mismo en que el usuario deja de comunicarse para que en un momento inmediato se entregue la información de navegación.

## 2.2. CONSIDERACIONES EN EL RECEPTOR

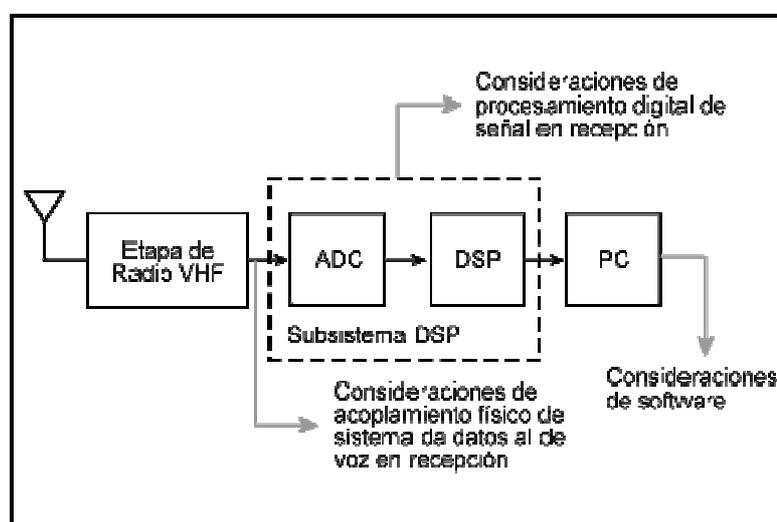


Figura 2. 17: Esquema básico estructural y consideraciones en recepción

- **Consideraciones de procesamiento digital de señal en recepción:** Establece los lineamientos que deben seguirse en la implementación de los módulos a ejecutarse por el microcontrolador DSP, destacándose en esta parte el filtrado digital y el proceso de demodulación de la forma de onda enviada por la contraparte transmisora para rescatar la información implícita, dando origen así al receptor de información de navegación.
- **Consideraciones de acoplamiento físico del sistema de datos al de voz en recepción:** Establece lo que hay hacer para adaptar de manera física el sistema

DSP representado en el Kit de desarrollo, a la radio que hará la recepción, para establecer así el prototipo físico del receptor.

- **Consideraciones de software:** Establece los lineamientos que deben tomarse en cuenta para aprovechar la información de navegación receptada y convertirla en un formato inteligible a través de un sistema de información geográfica agradable al usuario.

### 2.2.1. ACOPLAMIENTO FÍSICO DE VOZ Y DATOS EN RECEPCIÓN

Para la implementación física del prototipo se debe acoplar la tarjeta de desarrollo Ez Kit Lite a la radio base receptora, se debe considerar que la recepción de voz se la realiza a través de un parlante, lo cual es lógico considerando el hecho de que es un sistema netamente pensado para el manejo de audio, pero para un sistema en que se recibirán datos modulados en frecuencias audibles, el uso de un altavoz no tiene sentido, ya que la información recibida debe atravesar una etapa de demodulación en la que se recupere la información de navegación enviada por el transmisor.

En los equipos a utilizar, como lo mostrará la figura 2.18 existe la manera de guiar la información recibida hacia otro destino que no sea el altavoz interno, por ejemplo unos parlantes externos, esta salida debe ser aprovechada para la conexión con la tarjeta de desarrollo, de modo que la información de llegada pueda ser procesada por el sistema DSP.

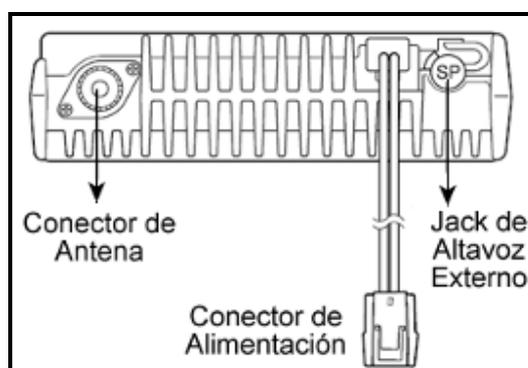
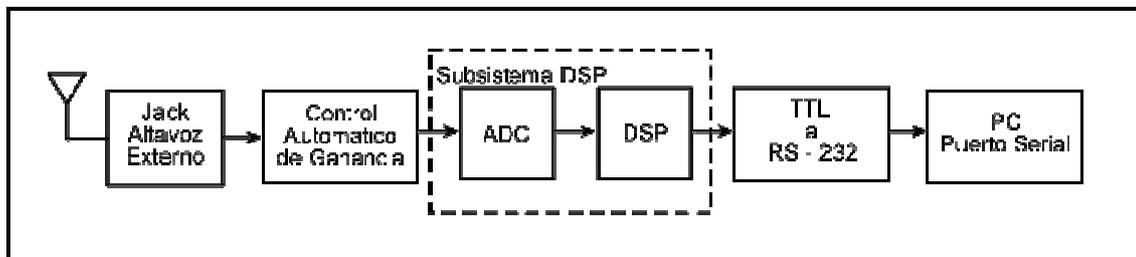


Figura 2. 18: Panel posterior de una radio base Kenwood

La conexión directa, del jack de altavoz externo de la radio receptora al jack del ADC de la tarjeta de desarrollo, tiene una dificultad, ya que el rango en amplitud de las señales entrantes a dicho ADC solo pueden estar en cierto intervalo, por debajo del cual la recepción es nula y por encima del mismo se produce saturación, lo que produce errores en la información recibida, se debe entonces construir un circuito de control automático de ganancia que permita mantener el nivel de la señal de entrada a la tarjeta en un rango aceptable por la misma. Esta precaución debe tomarse por cuanto las radio bases poseen controles de volumen que afectan el nivel en la salida de audio.



**Figura 2. 19:** Esquema de Acoplamiento físico en Receptor

### 2.2.2. PROCESAMIENTO DIGITAL DE SEÑAL EN RECEPCIÓN

Al igual que en transmisión, el procesamiento digital de señal en el receptor se lo ha dividido en módulos de tal manera que cada uno se encargue de una tarea específica, facilitando de esta manera el desarrollo, depuración y posterior entendimiento del sistema en discusión. A continuación se presenta un esquema ilustrativo de lo que deberá realizarse en el microcontrolador DSP y además como este interactúa con el exterior obteniendo datos y generando señales de salida.

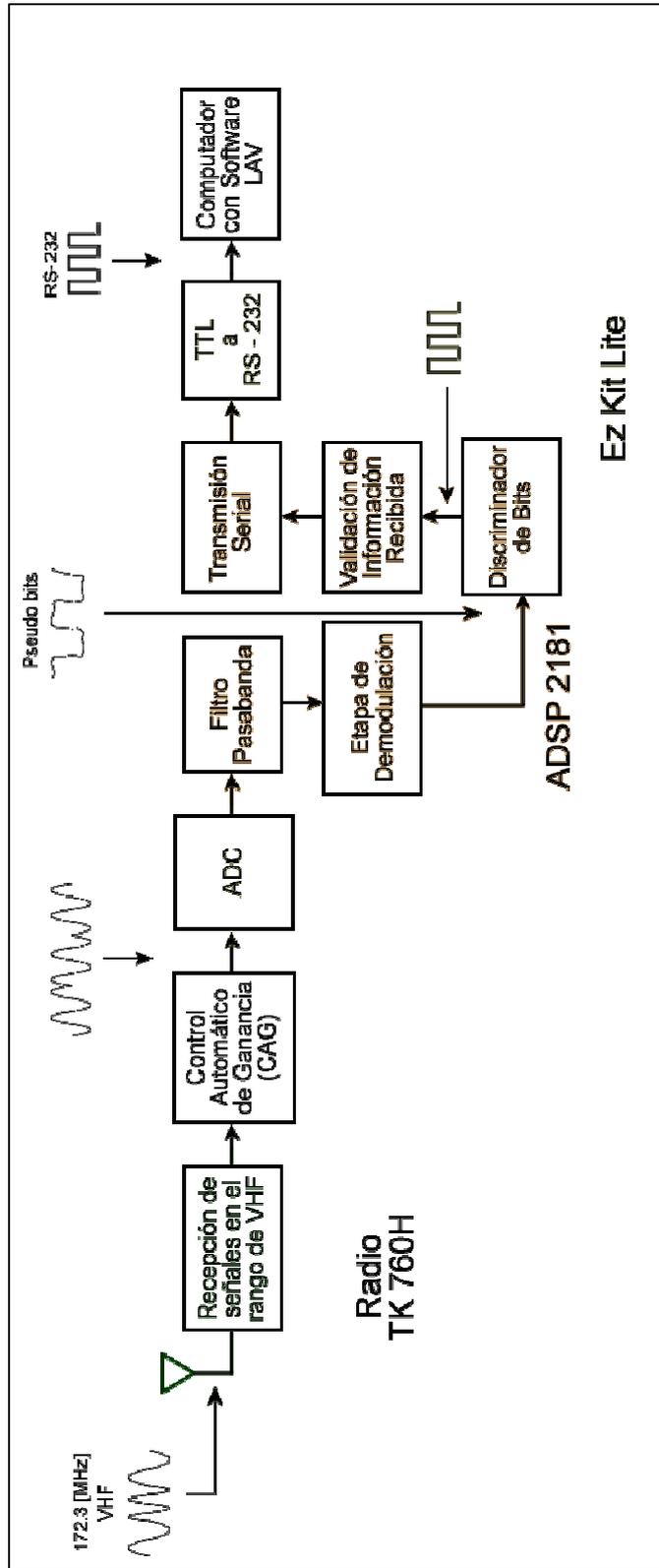


Figura 2. 20: Conformación del Receptor

En la figura 2.20 se puede apreciar claramente los módulos a ejecutarse en el microcontrolador DSP, a continuación se detalla la labor que debe cumplir cada etapa de procesamiento digital.

#### **2.2.2.1. Módulo de Filtrado**

Este módulo implementa un filtro pasa banda cuya principal función es separar la parte del espectro de la señal recibida que realmente sirve para efectos de demodulación, desechando de esta manera la parte del espectro de la señal recibida que pudo haber sido afectada por el ruido del canal y que no tiene ninguna importancia en el proceso siguiente.

#### **2.2.2.2. Módulo de demodulación**

Por lo general la etapa de demodulación es más complicada de implementar que su contraparte moduladora, ya que la primera debe incluir una serie de etapas que incluyen filtros digitales en tiempo real, lo que desde ya implica la necesidad de poseer una capacidad de procesamiento mediante una arquitectura adecuada. La etapa de demodulación también incluye una fase de discriminación de bits en la señal cuasi digital resultante de etapas anteriores.

#### **2.2.2.3. Módulo de discriminación de bits**

Esta etapa complementa la labor del demodulador poniendo a su salida los bits modulantes, esta etapa debe ser diseñada para ser inteligente, por cuanto la señal que tiene a su entrada es de tipo cuasi digital, ya que ha sido limitada en banda para poder ser transmitida y además ha sido afectada por las irregularidades del canal inalámbrico de comunicaciones.

Entonces esta etapa determina el estado lógico de los cuasi bits, típicamente respecto a un valor tomado como umbral, por encima del cual se considera uno lógico, caso contrario cero lógico.

#### **2.2.2.4. Módulo de validación de la información recibida**

Esta es una etapa importante ya que determina la validez de la información recibida, concluyendo si puede o no pasar a la siguiente etapa de procesamiento, en definitiva se trata de un control de calidad efectuado tomando como unidad de información la trama, es decir una vez demodulada y discriminada una trama completa se debe proceder a chequear su integridad.

#### **2.2.2.5. Módulo de transmisión serial**

Constituye la etapa interfaz entre el sistema receptor de información de navegación y el sistema encargado de hacer inteligible la información recibida, para que el usuario entrenado en la utilización del sistema LAV pueda hacer uso adecuado de la misma. Esta información se debe transmitir hacia el computador a una tasa razonable, generalmente el estándar en los programas terminales es de 9600 bps y es el que será adoptado para implementar esta etapa.

### **2.2.3. CONSIDERACIONES DE SOFTWARE.**

Como lo describe la figura 2.20, una vez que se demodula la información de navegación enviada por el transmisor del vehículo se ha recuperado la trama de navegación, desde este punto es tarea de la sección de software realizar las respectivas acciones que harán posible su aprovechamiento:

- Graficar la posición exacta del vehículo el momento de la transmisión de la información a través del software SIG Google Earth, presentando además información relativa al instante en el cual se capturó esa información y la velocidad que llevaba en dicho momento. Todo lo cual permite tomar decisiones en tiempo real.
- Almacenar en una base de datos de Microsoft Access el histórico posicional de todos los campos provistos en la trama receptada, de modo que posibilite un análisis posterior del comportamiento vehicular, pudiendo tomar decisiones

principalmente de retroalimentación que conduzca a un mejoramiento de las actividades realizadas.

Es evidente que el Google Earth y Microsoft Access no pueden realizar directamente la toma serial de la trama entregada por el demodulador, se debe entonces implementar en software un programa que permita recibir serialmente la trama enviada y establecer las condiciones necesarias para aprovechar cada uno de sus campos realizando de esta manera las acciones especificadas en cada uno de los puntos anteriores.

### **2.3. ESTRUCTURA DE DATOS A USAR**

Establece la manera como se organizan los datos en el sistema procesador de información, de tal manera que exista coherencia entre emisor - receptor y no se produzcan errores debido a fallas de sincronización.

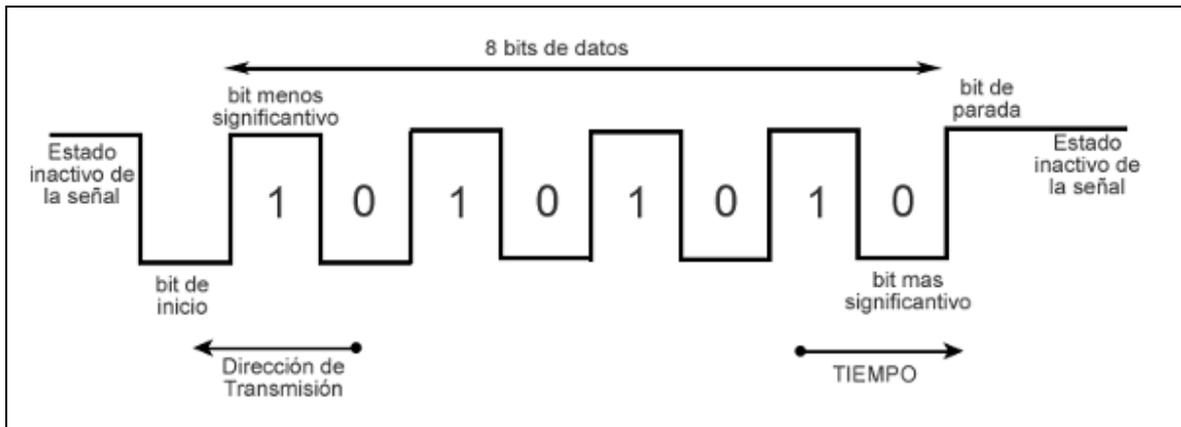
#### **2.3.1. MODO DE TRANSMISIÓN ASÍNCRONO**

El modo de transmisión asíncrono considera de antemano que no existe una sincronización previa entre transmisor y receptor dada por una señal de reloj compartida por ambas terminales, sino que este proceso de sincronización se da cada vez que inicia la recepción de un carácter.

Si bien el modo de transmisión síncrono posee un mayor rendimiento, el modo asíncrono ofrece el suficiente soporte para el envío de información de navegación, ya que esta es de baja velocidad de señal y se transmite de forma no continua, además el modo asíncrono es de fácil implementación, en contraste, el modo síncrono es usado en velocidades altas de modulación, mayores a 1200 baudios, en aplicaciones en las cuales el envío de datos es considerado continuo y su implementación presenta una mayor complejidad.

### 2.3.2. ESTRUCTURA DE CARACTER

El modo de transmisión asíncrono a utilizar emplea la siguiente estructura de carácter:



**Figura 2. 21:** Carácter asíncrono

Una de las principales ventajas del modo asíncrono es que, en el supuesto caso no consentido, exista un error de sincronización este no afecta una gran cantidad de bits de datos, sino que los errores de este tipo se focalizan en un carácter determinado, ya que la sincronización se da tomando como unidad el carácter.

En principio la línea serial se encuentra en estado inactivo (idle) esperando por un cambio de estado (bit de inicio) que le indique que los próximos bits son considerados datos válidos, el sistema de procesamiento contabiliza los ocho bits de datos y espera un retorno al estado inactivo (bits de parada), en modo asíncrono el primer bit en ser transmitido es el menos significativo, terminando con el más significativo.

Es opcional la inclusión de un bit de paridad con la intención de realizar un primitivo control de errores, de darse el caso este se ubicaría al final de los bits de datos, en el presente proyecto se decide no ubicarlo, ya que como se mencionó, se utilizará otro mecanismo de control.

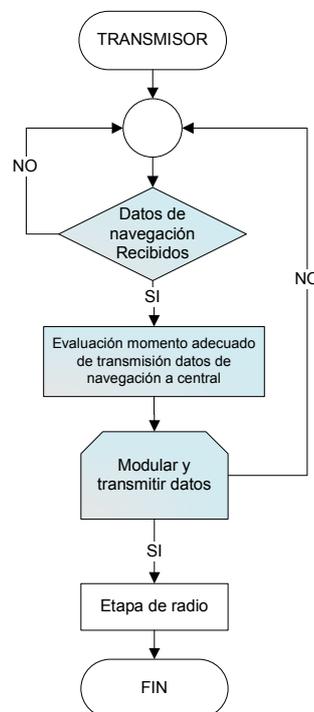
## CAPÍTULO 3

### IMPLEMENTACIÓN DEL PROTOTIPO

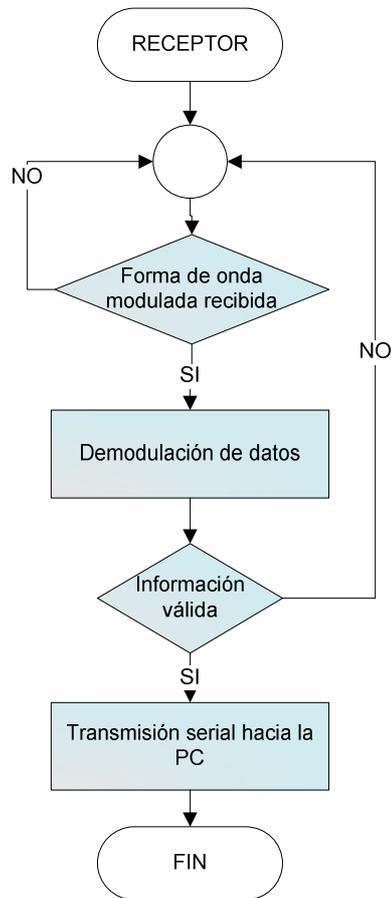
El presente capítulo pone en marcha las estrategias establecidas en el capítulo anterior, tomando en cuenta los recursos de los cuales se dispone, sus limitaciones y ventajas, se hace referencia a diferentes sub etapas que conforman tres principales componentes, y a como cada uno de estos se integra para formar un sistema de localización automática vehicular completamente funcional y útil, montado sobre un sistema concebido inicialmente para operar solamente con información relativa a voz.

#### 3.1. FIRMWARE

Constituye el primer componente principal del sistema LAV total, ya que este permite manejar los datos de navegación provenientes de la infraestructura GPS, de tal manera que se los adapte al canal de comunicaciones inalámbrico del cual se dispone y además maneje de manera inteligente la convivencia con el sistema de voz, sobre el cual se monta el sistema de localización vehicular.



**Figura 3. 1:** Esquema básico de procesos firmware transmisor



**Figura 3. 2:** Esquema básico de procesos firmware receptor

Los diagramas de flujo anteriores describen lo más básico posible los procesos que deben realizarse en los sistemas DSP fundamentalmente en el microcontrolador DSP.

A continuación se detalla la implementación de cada uno de los módulos que conforman el firmware del transmisor y receptor, de acuerdo a las directivas establecidas en los numerales respectivos de procesamiento digital de señal en el capítulo segundo.

En el Anexo III se detalla el código fuente de cada uno de los módulos que componen el *Firmware*, de tal manera que se puede realizar una rápida consulta si se cree conveniente.

### 3.1.1. TRANSMISOR

En la figura 3.1 se presentó un esquema simple de los procesos a realizarse, a continuación se presenta uno más elaborado con el fin de guiar la comprensión de dichos procedimientos.

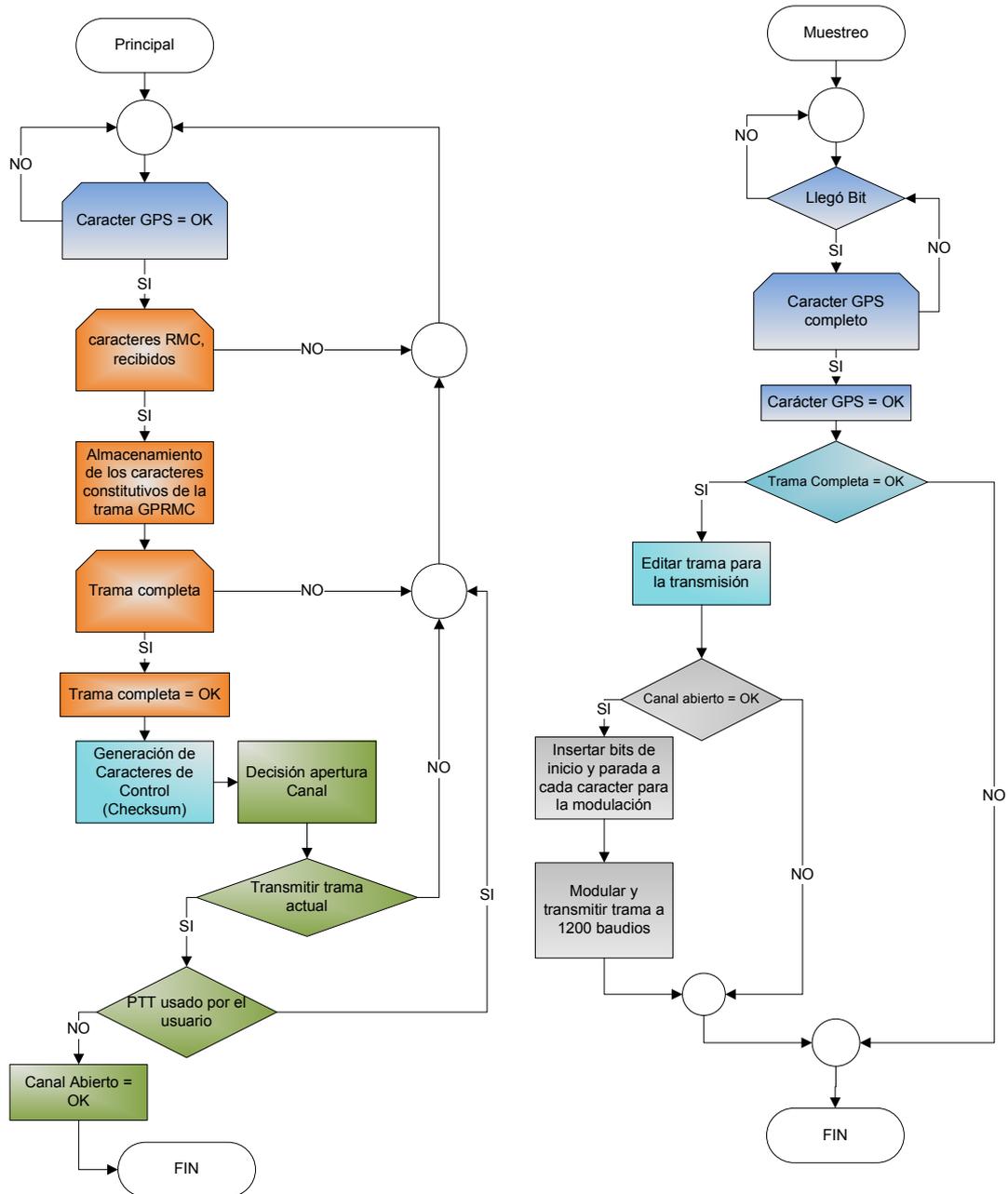


Figura 3. 3: Diagrama de flujo general de transmisión

	Módulo de recepción y almacenamiento de caracter desde el GPS
	Módulo de detección y almacenamiento trama GPRMC
	Módulo de edición de trama almacenada
	Módulo de evaluación de apertura automática de canal
	Módulo de generación de onda modulada

Como puede notarse en la figura 3.3 los módulos que conforman el procesamiento digital de señal en el transmisor se implementan tanto en el programa *principal* como en la rutina de servicio a la interrupción de *muestreo*, cuales son las que se implementan en una sección en particular no es una decisión arbitraria, las que se encuentran en *muestreo* necesitan de una base de tiempo fija para poder operar, dicha base temporal constituye el período de muestreo.

A continuación se pormenoriza cada uno de los módulos que realizan el procesamiento digital de señal en el transmisor, se describen las etapas de modo que vayan en un orden lógico ascendente, es decir desde la captura de la información de navegación hasta la transmisión de la misma, primeramente se ubica un diagrama de flujo con el procedimiento detallado y posteriormente se realiza una descripción del algoritmo implementado que trata de mejorar el entendimiento de dicho diagrama de flujo y el proceso que representa.

### **3.1.1.1. Módulo de recepción y almacenamiento de caracter desde el GPS**

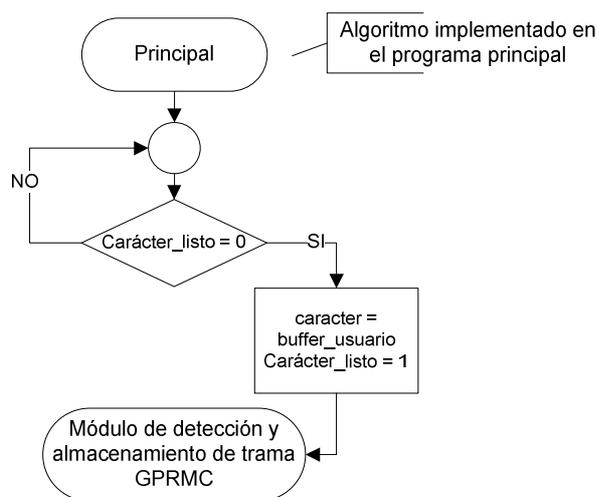
Según lo señalado en el numeral 2.1.4.2<sup>45</sup> del capítulo segundo, esta etapa debe implementar un software UART (Universal Asynchronous Receiver - Transmitter) que

---

<sup>45</sup> Página 93 del presente proyecto de titulación

permita la captura de datos (carácter a carácter) provenientes de la infraestructura GPS a la velocidad NMEA<sup>46</sup>.

Como puede visualizarse en la figura 3.3 este módulo se encuentra implementado tanto en la sección denominada *principal*, como en la rutina de servicio a interrupción llamada *muestreo*, el siguiente gráfico representa la parte que se está en principal.



**Figura 3. 4:** Diagrama de flujo almacenamiento carácter GPS en principal

La parte *principal* de este módulo espera a que se haya almacenado un carácter completo proveniente del dispositivo GPS, de esta manera guarda el carácter en otra localidad y activa la bandera respectiva que da paso al análisis del byte en el siguiente módulo de procesamiento.

La segunda parte de este módulo es la encargada del ensamblado bit a bit del carácter anteriormente mencionado y como se dijo se implementa en la rutina de servicio a la interrupción de *muestreo*, ya que requiere tener una base de tiempo para operar, su respectivo diagrama de flujo se describe a continuación:

<sup>46</sup> 4800 bps

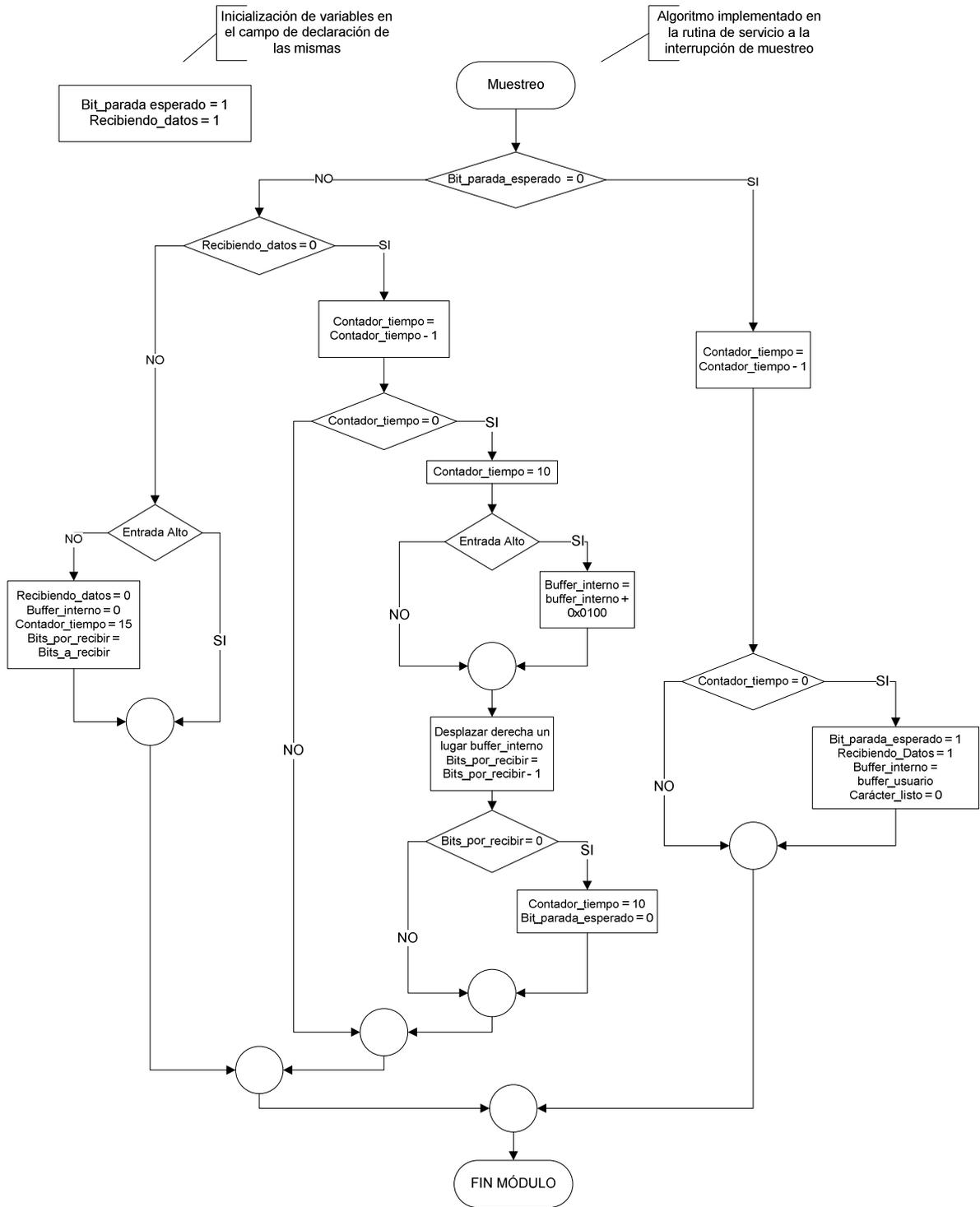


Figura 3. 5: Diagrama de flujo almacenamiento GPS en muestreo

Esta parte del módulo realiza bit a bit la adquisición de los caracteres que se envían a través de un dispositivo GPS, los cuales son transmitidos desde este hacia el microcontrolador ADSP 2181 a través del puerto serial del Ez Kit Lite a una velocidad de 4800 bps; estos son expedidos y receptados de forma asíncrona cuyas características son: ocho bits de información uno de inicio y otro de parada.

Este módulo está diseñado expresamente para funcionar a esta velocidad de transmisión y a una frecuencia de muestreo del ADC – DAC (*Códec*) de 48000 Hz, Para lograr la velocidad de recepción, se toma como base de tiempo el período de muestreo:

$$T = (48000\text{Hz})^{-1} = 20.83 * 10^{-6}\text{seg}$$

Cada  $T$  segundos se produce una interrupción, en la rutina de servicio a la misma se debe verificar la posición de las banderas que indican el estado serial de la muestra de la señal analizada, pero no en cada interrupción sino cada diez de tal modo que:

$$10 * T = 20.83 * 10^{-5}\text{seg}$$

Resulta ser el tiempo de bit, cuyo inverso:

$$(20.83 * 10^{-5})^{-1} = 4800 \text{ bps} = V_{tx}$$

Todo esto explica el valor máximo de la variable *contador\_tiempo*, que siempre es diez, exceptuando el caso en que se ha detectado el bit de inicio, momento en cual se la iguala a quince, esto se justifica por cuanto se debe dejar pasar el bit de inicio y además alinearse con el centro del primer bit de información. Al entrar al servicio de interrupción *Muestreo*, se verifica inicialmente si es que se está esperando por el bit de parada, si no, el proceso puede estar recibiendo bits ó esperando por el bit de inicio.

Para comenzar el procedimiento, las banderas, *Bit\_parada\_esperado* y *recibiendo\_datos* deben estar inicializadas en uno lógico, para que sea posible entrar directamente a detectar el bit de inicio; una vez detectado el mismo se pone en cero lógico la bandera *recibiendo\_datos*, se inicia un contador fijado en 15 que al expirar hace la primera comprobación del bit mediante la muestra tomada en ese instante,

es un cero o un uno, se fija además el número de bits a recibir y se pone en cero el buffer interno donde se irán almacenando los bits conforme se van detectando.

Luego de la primera comprobación de bit se fija el contador en 10, ya no en 15, se disminuye en uno la cantidad de bits por recibir, al expirar el nuevo valor del contador se realiza la discriminación del bit, luego del cual se lo almacena, se vuelve a cargar el contador con el valor de 10, se disminuye en uno los bits a recibir y así sucesivamente hasta que este último valor sea cero, luego de ello se pone en cero la bandera *bit\_parada\_esperado*, se carga con 10 el contador, se espera que finalice este conteo, luego del cual el carácter está totalmente listo, encendiendo la bandera de *caracter\_listo* y transfiriendo el contenido del buffer interno al buffer de usuario, se vuelve a cargar las banderas *bit\_parada\_esperado*, *recibiendo\_datos* con cero lógico, el proceso entonces se reinicia en espera de un nuevo carácter.

Como se explicó en el programa principal se verifica el estado de la bandera *caracter\_listo*, cuando esta toma el valor bajo se transfiere el contenido del buffer de usuario a una variable denominada *carácter* que será usada en el siguiente módulo de programa, finalmente se pone uno a la variable *carácter\_listo* en espera de que otro carácter esté totalmente ensamblado y listo para usarse.

### **3.1.1.2. Módulo de detección y almacenamiento trama NMEA GPRMC**

Según lo señalado en el numeral 2.1.4.3<sup>47</sup> del capítulo segundo, esta etapa debe verificar, en base al carácter capturado por el modulo anterior, si se trata de la secuencia RMC perteneciente a la trama GPRMC, con lo cual se la identifica y se procede a almacenarla. Como puede apreciarse en la figura 3.3 este módulo está completamente implementado en la sección *principal* de programa, por lo que no necesita de una base de tiempo en específico para operar.

A continuación se presenta el diagrama detallado que describe el funcionamiento del módulo:

---

<sup>47</sup> Página 93 del Presente proyecto de titulación

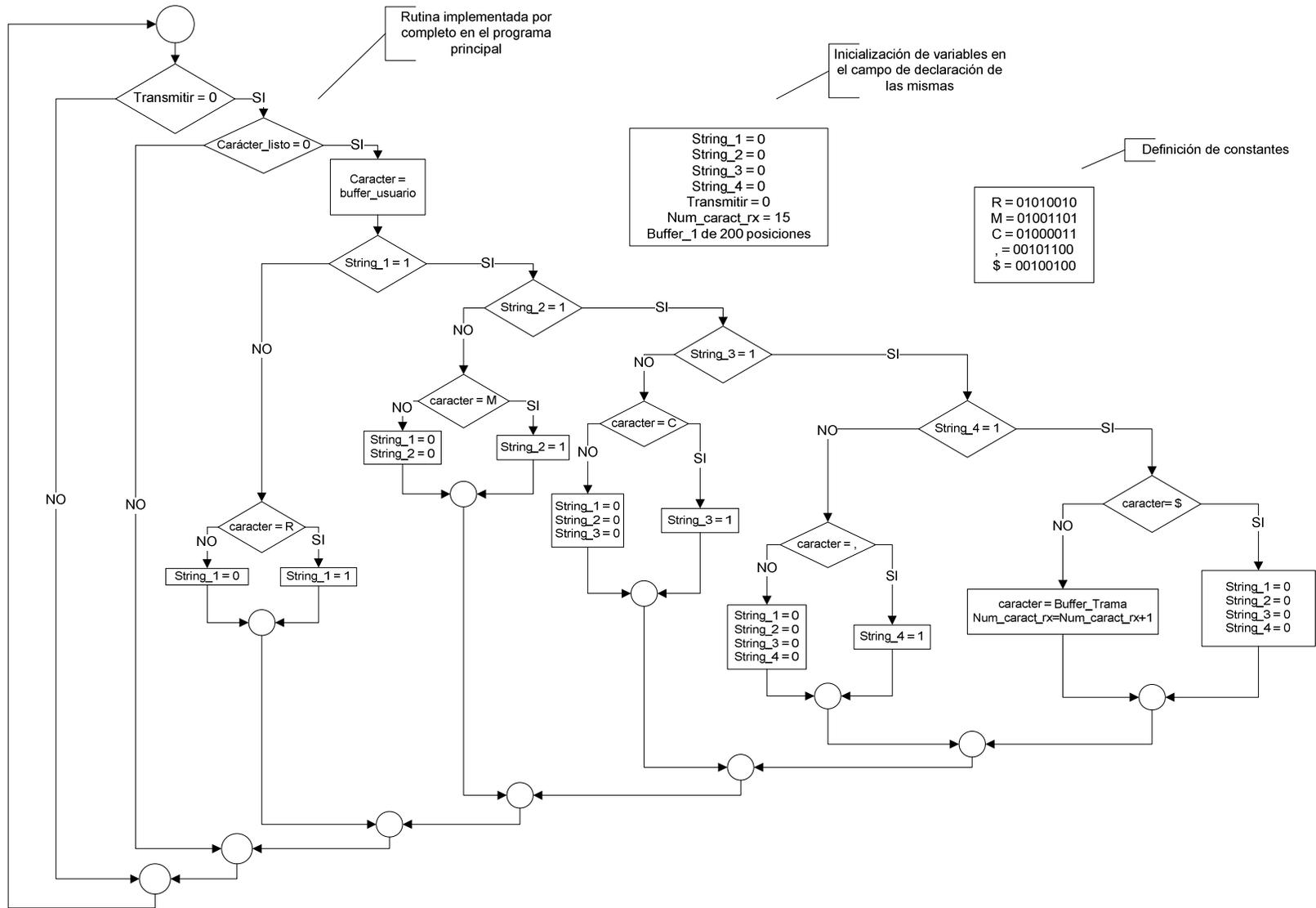


Figura 3. 6: Diagrama de flujo detección y almacenamiento trama NMEA GPRMC

El proceso de reconocimiento y almacenamiento de la trama GPRMC, se lo hace inmediatamente después de que el módulo anterior entrega el caracter recibido desde el dispositivo GPS, este procedimiento se lo realiza en el programa principal y se lo detalla a continuación:

Una vez que se ha obtenido el caracter, comienza una serie de comparaciones, la primera es saber si dicho caracter es R, si lo es, se pasa a buscar M, si no, se reinicia la búsqueda de R, si se trata de M, se busca C, si no se reinicia la búsqueda de R y así sucesivamente hasta haber hallado RMC, es necesario también buscar después de C el carácter (,) ya que después de este comienza la información útil, las banderas *string* indican si el carácter correspondiente se ha recibido o no, para de este modo continuar con la búsqueda del siguiente o reiniciar la búsqueda desde el primero según sea el caso.

Luego de haber hallado RMC(,) se procede a guardar los datos en un buffer circular, este procedimiento se detiene cuando se encuentra el carácter \$, este representa el inicio de otra trama NMEA y no se lo almacena, la razón para buscarlo es que de esta manera se puede guardar toda la trama GPRMC incluyendo los caracteres carrier return ( $\backslash n$ ) y line feed ( $\backslash r$ ).

El buffer circular donde se guardan los datos, inicialmente no se halla totalmente vacío, este tiene doscientas posiciones de las cuales quince son pre cargadas y contienen información fija, desde la 185 a la 192, contienen caracteres de entrenamiento cuyo valor en binario es 11111111, la posición 193 es ocupada por un carácter que representa un espacio en blanco, la posición 194 por el carácter > que indica el inicio de la trama a transmitir, las posiciones 195 a 198 son configuradas con el nombre del dispositivo móvil, en este caso DSP1, finalmente la posición 199 es cargada con el carácter coma (,) indicando el fin del campo "*nombre del móvil*" y el inicio de información obtenida del dispositivo GPS que será almacenada desde la siguiente posición identificada como número 0.

Todos estos caracteres pre cargados serán transmitidos como parte de la trama, esto explica que se inicialice la variable *num\_caract\_rx* con un valor de quince y que, a partir de este se empiece a incrementar los caracteres recibidos, el valor máximo al que llegue esta variable será usada en la etapa de modulación de bits y será interpretada como el número de caracteres a modular y transmitir.

El módulo que se discute inicia con la comprobación de una bandera denominada *transmitir*, esta es manejada por una etapa siguiente para iniciar o suspender el proceso de modulación y transmisión de la trama almacenada; en esta parte, si tiene un valor alto indica que en ese instante se está transmitiendo una trama previamente almacenada y por tanto se suspende el proceso de búsqueda y almacenamiento de una trama entrante hasta que haya finalizado la transmisión.

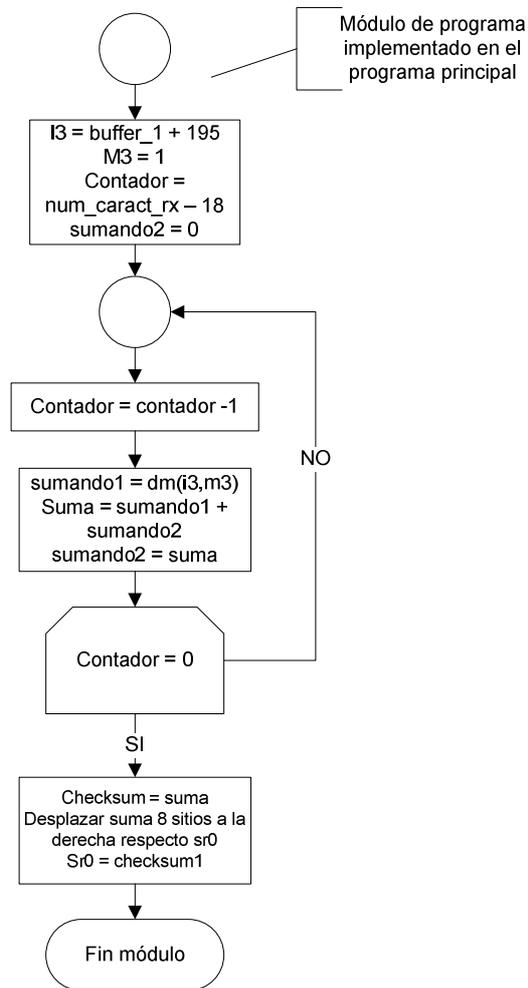
### **3.1.1.3. Módulo de edición de trama almacenada**

Según el numeral 2.1.4.4<sup>48</sup> del capítulo segundo este módulo debe hacer posible la edición de la trama GPRMC almacenada por el módulo anterior, lo que incluye la generación de un nuevo checksum, adición del nombre del móvil (*DSP1*) y caracteres de inicio y fin de trama, así como también la eliminación de campos innecesarios.

Como puede visualizarse en la figura 3.3 este módulo posee dos partes plenamente identificadas, la primera es el procedimiento por el cual se generan los caracteres de control que implementan el campo *checksum*, lo cual se lo hace en la sección *principal* de programa, no necesitando de una base de tiempo para su funcionamiento, tal como lo describe la siguiente figura:

---

<sup>48</sup> Página 94 del Presente proyecto de titulación



**Figura 3. 7:** Diagrama de flujo generación de caracteres de control en transmisión

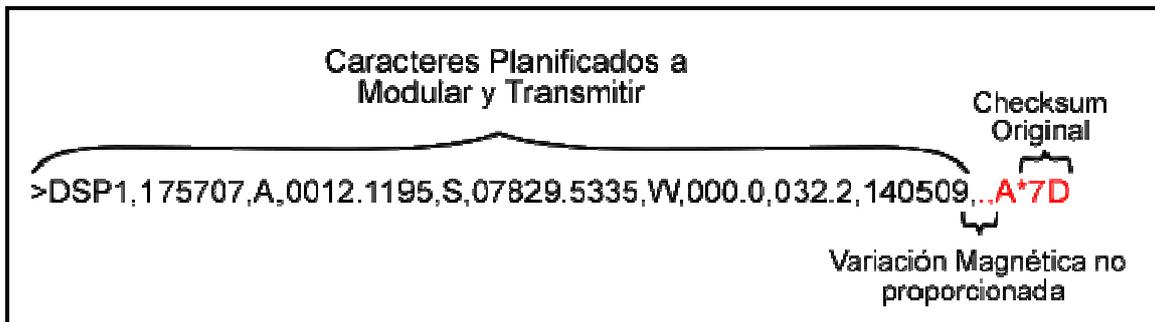
El cálculo de los caracteres de control que implementan el checksum se lo realiza a partir del primer carácter del campo “*nombre del móvil*”, por ello se ubica el puntero del buffer circular, que contiene dichos datos almacenados en la posición 195. Un contador almacena el número de sumas consecutivas a realizar, dado este por la resta entre el número de caracteres totales en el buffer (*num\_caract\_rx*) y el número de caracteres que no se tomarán en cuenta en el cálculo del checksum, estos son los de entrenamiento, el espacio en blanco, el símbolo >, carrier return, line feed, el campo de desviación magnética (por lo general no provisto) y sus dos respectivas comas separadoras de campo, el checksum original y el asterisco; todo en total suma 18 caracteres no tomados en cuenta en el cálculo del checksum, estos se los ignora

por cuanto la información que se considera importante está entre los campos “*nombre del móvil*” y “*fecha*”.

Se realiza la suma como se indica en el diagrama de flujo, cuando ésta ha terminado se guarda el total en una variable llamada *checksum*, la suma por lo general ocupa 16 bits, los 8 primeros de la derecha representarán el carácter menos significativo y los 8 siguientes representarán el carácter más significativo del checksum, por lo que para obtener este último se debe desplazar la suma total 8 lugares hacia la derecha tomando como referencia el registro sr0, hecho esto el carácter más significativo del checksum puede ser tomado de este registro y almacenado en una variable denominada *checksum1*.

La segunda parte de este módulo completa la edición propiamente dicha de la trama original, para lo cual elimina el campo de variación magnética, que no es provisto en la práctica y el campo de checksum original, todo lo cual es reemplazado por el checksum calculado con la primera parte del módulo en discusión y con los caracteres de fin de trama <nl>.

La parte mencionada se implementa en la sección de programa *muestreo* y el momento que se la aplica corresponde al final del proceso de modulación, es decir cuando se hayan modulado y transmitido todos los caracteres planificados de trama, indicados por la figura 3.8, se eliminan entonces los campos mencionados y se reemplazan por los caracteres de fin de trama que son inmediatamente modulados y transmitidos.



**Figura 3. 8:** Trama GPRMC semi – editada

Al realizarse este procedimiento junto con el de modulación es conveniente que su diagrama de flujo y correspondiente explicación sean descritos en un momento oportuno más adelante.

#### **3.1.1.4. Módulo de evaluación de apertura automática de canal**

Esta etapa es fundamental en el sistema de transmisión ya que permite abrir el canal para que los datos modulados puedan ser depositados en el, así como también provee la suficiente inteligencia a dicho sistema para decidir los momentos en los que se debe transmitir la información de navegación, tal como se lo planteó en el numeral 2.1.4.5<sup>49</sup> del capítulo segundo.

Como se puede apreciar en la figura 3.3 este módulo se lo implementó por completo en el programa principal por no necesitar de una base de tiempo específica para su operación, el diagrama de flujo y su explicación se detallan a continuación:

<sup>49</sup> Página 94 del presente proyecto de titulación

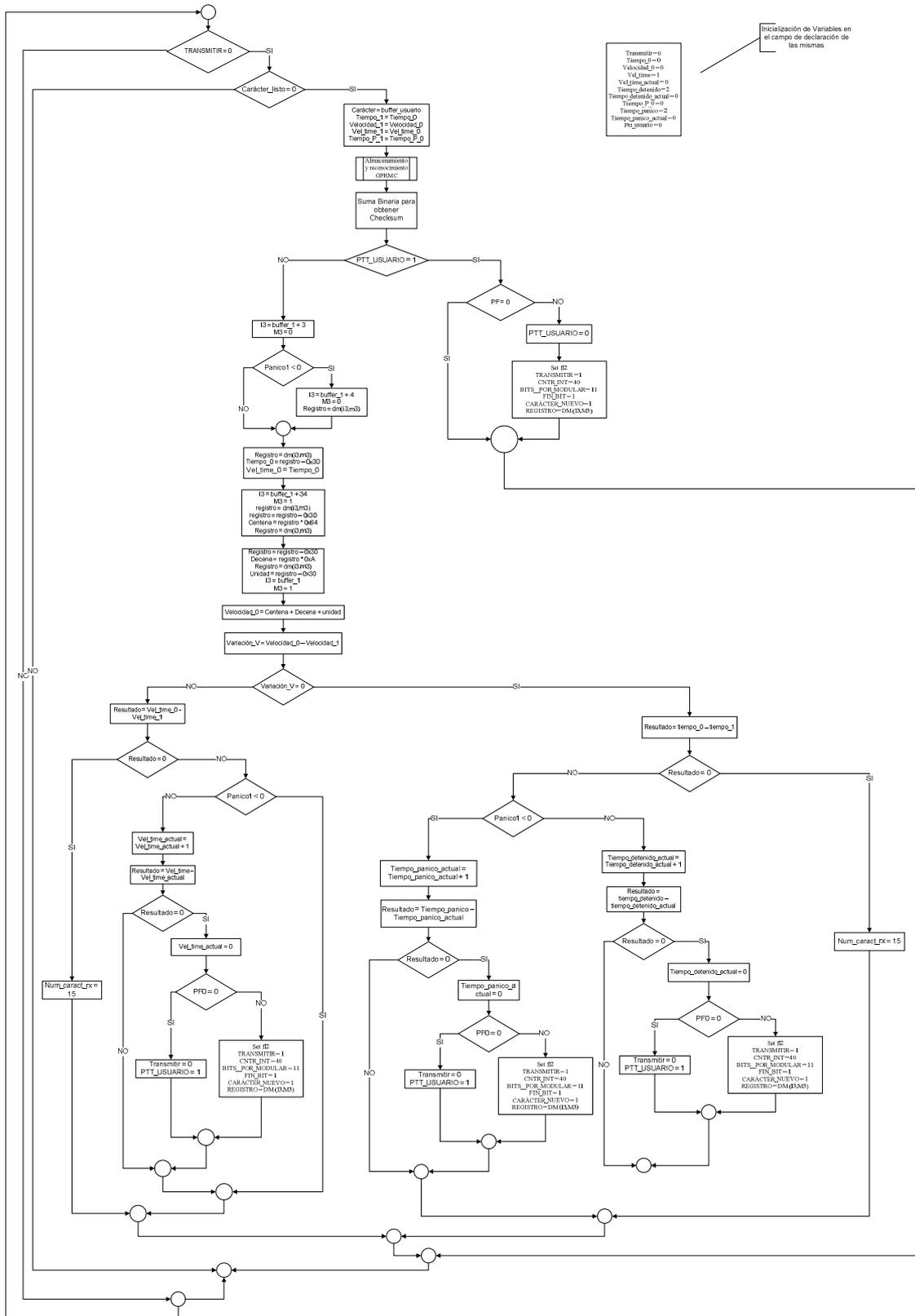


Figura 3. 9: Diagrama de flujo módulo de evaluación de apertura automática de canal

Una vez que las etapas de reconocimiento y almacenamiento de la trama GPRMC, así como la primera parte del módulo de edición de trama almacenada, se han cumplido, se entrega el control al módulo en discusión, el cual se lo puede dividir en dos partes.

La primera encargada de tomar la decisión de apertura del canal, con lo cual se da paso al proceso de modulación y transmisión de la información, esto en base al cumplimiento de ciertos criterios relacionados a variables de tiempo, velocidad y estado de pánico. La segunda encomendada de vigilar la activación del botón *push to talk*, usado en las comunicaciones normales por voz, con el objetivo de evitar que la transmisión de información de localización interrumpa inesperadamente las comunicaciones por voz, así mismo poder transmitir información de localización en el caso en que no haya podido hacérselo debido a comunicaciones de voz, esto inmediatamente después que el usuario deje de presionar el botón *push to talk*.

Ambas partes se las implementa de una sola vez, así que se tratará de elaborar una explicación lo suficientemente clara para su entendimiento.

La bandera *ptt\_usuario* se inicializa con cero, se considera entonces que en principio el botón de *push to talk* esta libre y se pasa a verificar las condiciones para efectuar la transmisión de información de navegación. La primera consideración consiste en averiguar si el usuario ha presionado el botón de pánico, lo cual se consigue verificando la bandera *panico1*, esta es actualizada con un valor positivo o negativo cada vez que se presiona dicho botón, a través de un proceso implementado en la rutina de servicio a la interrupción IRQE.

Se supone, por el momento, que el botón de pánico no se ha presionado por lo que *panico1* es positivo, se toma entonces la unidad de minuto del campo *hora* de la trama GPRMC previamente almacenada, para tomarla como base de tiempo, antes de poder operar con este valor se lo debe convertir a Hexadecimal.

Este valor ya utilizable se lo guarda en dos variables, la primera relacionada con el tiempo y la segunda con velocidad, *Tiempo\_0* y *Vel\_time\_0* respectivamente, esto se

explica por cuanto, en condiciones de movimiento y velocidad la base de tiempo es el minuto.

Posteriormente se ubica el puntero del buffer circular en la posición en que se guarda las unidades de velocidad para poder recuperar este valor y utilizarlo en el cálculo de la velocidad, este mismo procedimiento se realiza con las decenas y centenas del valor de velocidad, una vez obtenidas estas tres cantidades se procede a obtener el valor de la velocidad en formato decimal con el que se pueda operar.

Se tiene entonces el valor actual de velocidad (*velocidad\_0*), este se lo compara con el obtenido anteriormente (*Velocidad\_1*), determinando así la existencia o no de movimiento del vehículo, de resultar lo primero, se toma en cuenta la variable *vel\_time\_0* para compararla con *vel\_time\_1*, que guarda el valor anterior de *vel\_time\_0*, si son iguales no ha habido variación de minutos por lo que se retorna al programa principal. De existir variación significa que ha habido un cambio de minuto, se pasa entonces a incrementar un contador y a verificar si ha llegado al valor tope, este límite puede ser configurado en la variable *vel\_time* con el número de minutos que se quiere transcurrir hasta la próxima transmisión, de alcanzar el valor máximo así estipulado se inicia a cero los minutos transcurridos para transmitir, esto en la variable *vel\_time\_actual*.

El siguiente paso es verificar si el usuario está o no ocupando el canal con una comunicación de voz, esto último se lo hace sencillamente verificando el pin PF0, disponible en las ranuras de expansión del Ez Kit Lite, el que se lo conecta físicamente a la señal del *push to talk* del usuario.

Si PF es uno el canal esta libre y se procede a iniciar la modulación a través de la fijación de ciertos valores en correspondientes registros a ser usados por el siguiente módulo de procesamiento digital. Si PF es cero el canal está siendo usado por el usuario en una comunicación de voz se pone entonces uno en la bandera *ptt\_usuario* y se retorna al inicio del programa principal.

Ahora si el resultado de comparar *velocidad\_0* y *velocidad\_1* es cero implica que no hay movimiento y por tanto se pasa a un proceso idéntico al descrito en existencia de velocidad, con la diferencia que el valor tope en cantidad de minutos a esperar entre transmisión es diferente y se fija en la variable *tiempo\_detenido*, en lugar de utilizar *vel\_time\_0* y *vel\_time\_1*, se usa *tiempo\_0* y *tiempo\_1*, además de usar *tiempo\_detenido\_actual* en lugar de *vel\_time\_actual*.

En caso de entrar en pánico la bandera *panico1* es menor a cero y en lugar de apuntar hacia los minutos se lo hace hacia las decenas de segundo, se hace utilizable este valor y se lo guarda en las variables *vel\_time\_0* y *tiempo\_0*.

Sea con movimiento o sin este el estado de pánico debe mantenerse con transmisiones de la posición más seguidas, de esta manera en ambos procesos (con velocidad o detenido) se implementa la verificación de *panico1* para dar paso a verificar el tope máximo, en decenas de segundos, programado para transcurrir entre transmisión y transmisión, dicha programación se la hace en la variable *tiempo\_panico* y el contador que se incrementa se lo llama *tiempo\_panico\_actual*; como en los casos anteriores la voz tiene prioridad por lo que se pasa a verificar el estado del pin PF0 antes de pasar al proceso de modulación.

Si el usuario esta realizando una comunicación por voz, la transmisión de información de localización no es posible, en principio la trama que se iba a enviar se pierde, pero se compensa esta pérdida haciendo que se transmita apenas el usuario deje de presionar el botón *push to talk*, para ello cuando se detecta presionado dicho botón se enciende la bandera *ptt\_usuario*, haciendo que se verifique el pin PF0 la próxima vez que se entre a efectuar el presente módulo, buscando la condición cero que permita pasar inmediatamente al proceso de modulación sin necesidad de comprobar parámetros de velocidad o tiempo.

#### **3.1.1.5. Módulo de generación de onda modulada**

En el capítulo segundo se analizó la necesidad del uso de un buen esquema de modulación, se inició con la evaluación de posibilidades de tipos de modulación a

implementar en base a consideraciones teóricas, en este numeral se complementa con la adición de un elemento de juicio práctico que es la de implementación usando las capacidades del sistema DSP incluido en el Kit de Desarrollo, de tal manera que pueda realizarse un análisis final cuya conclusión sea el tipo de modulación definitiva a implementar.

En este numeral se usan los conceptos de generación de onda utilizando un microcontrolador DSP explicados en el numeral 1.3.3<sup>50</sup> del capítulo primero.

#### **3.1.1.5.1. Modulación ASK**

En modulación la generación de forma de onda modulada respecto a los tiempos de los bits modulantes requieren del uso de una base de tiempo fija por lo cual, como puede visualizarse en la figura 3.3 la etapa de modulación se la implementa en la rutina de servicio a interrupción de *muestreo*.

Como se explicó brevemente en el capítulo dos, en la modulación ASK la portadora se enciende durante un uno lógico, para el tiempo de bit de un cero lógico la portadora se suprime, tal como se muestra en la figura 2.8, a continuación se describe el diagrama de flujo respectivo como su explicación:

---

<sup>50</sup> Página 35 del presente proyecto de titulación

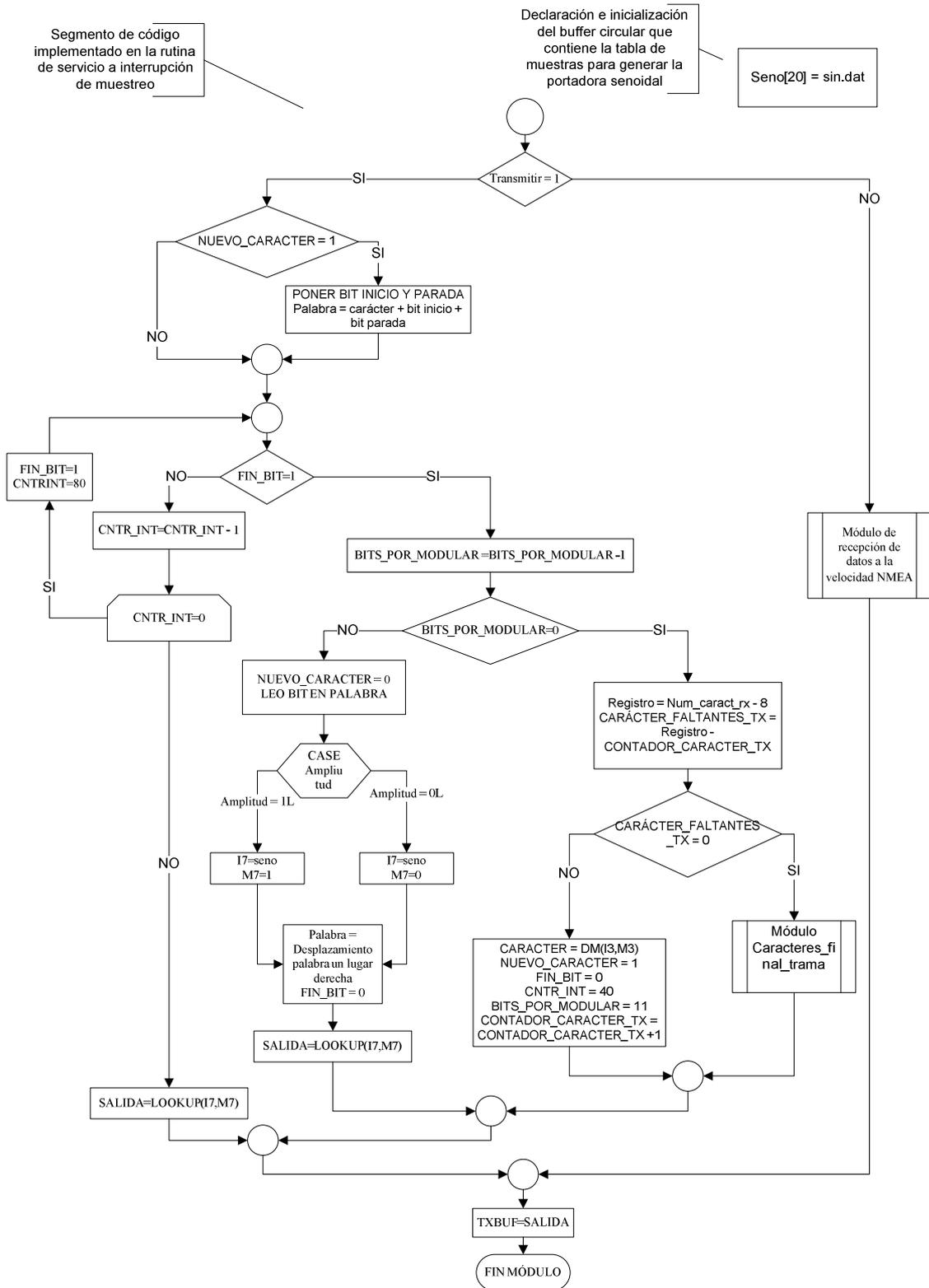
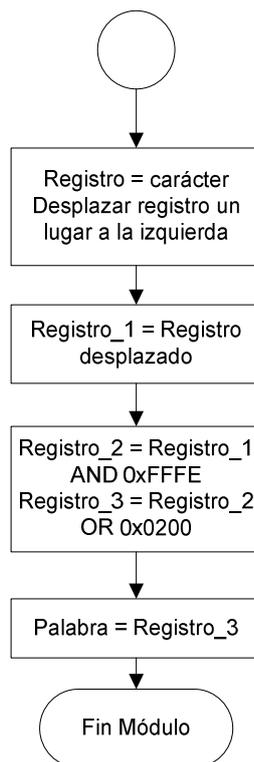


Figura 3. 10: Diagrama de flujo modulación ASK

Una vez realizados todos los procedimientos anteriormente explicados se espera a que el módulo de evaluación de apertura automática del canal autorice el envío de la información de navegación, activando la bandera transmitir.

Así entonces el procedimiento comienza verificando la bandera transmitir, si esta ha sido seteada a un valor de uno lógico se inicia el proceso de modulación de la trama previamente almacenada.

A continuación se verifica la bandera nuevo\_carácter, si esta es igual a uno significa que un nuevo carácter ha de ser procesado, de lo contrario actualmente ya se está procesando uno, dicho proceso consiste primeramente en acoplarle los bits de inicio y parada con una tarea descrita por el siguiente diagrama de flujo:



**Figura 3. 11:** Diagrama de flujo inserción bits de inicio y parada

Los ocho caracteres que componen el byte de información están contenidos en la variable *carácter*, a esta se le realiza un desplazamiento para dar origen al lugar que

ocupará el bit de inicio, finalmente se realizan dos operaciones lógicas que insertan el bit de inicio y parada respectivamente.

Los bits a modular están contenidos finalmente en la variable *palabra*, se inicia entonces el proceso de modulación bit a bit, en esta parte se hace uso de la bandera *fin\_bit*, cuando esta sea uno lógico habrá terminado el proceso modulador para un bit, y se actualiza un contador descendente que indica el número de bits que faltan por modular (*bits\_por\_modular*); si *fin\_bit* es cero quiere decir que se está modulando el bit actual.

El procedimiento de modulación empieza con el bit de inicio seguido por los bits de información en orden ascendente de significancia hasta terminar con el bit de parada.

Como se mencionó en el primer párrafo del presente numeral, el procedimiento de modulación se lo efectúa en la rutina de servicio a la interrupción de *muestreo*, lo cual es necesario por cuanto se necesita de una base de tiempo fija para conseguir la velocidad de señal deseada.

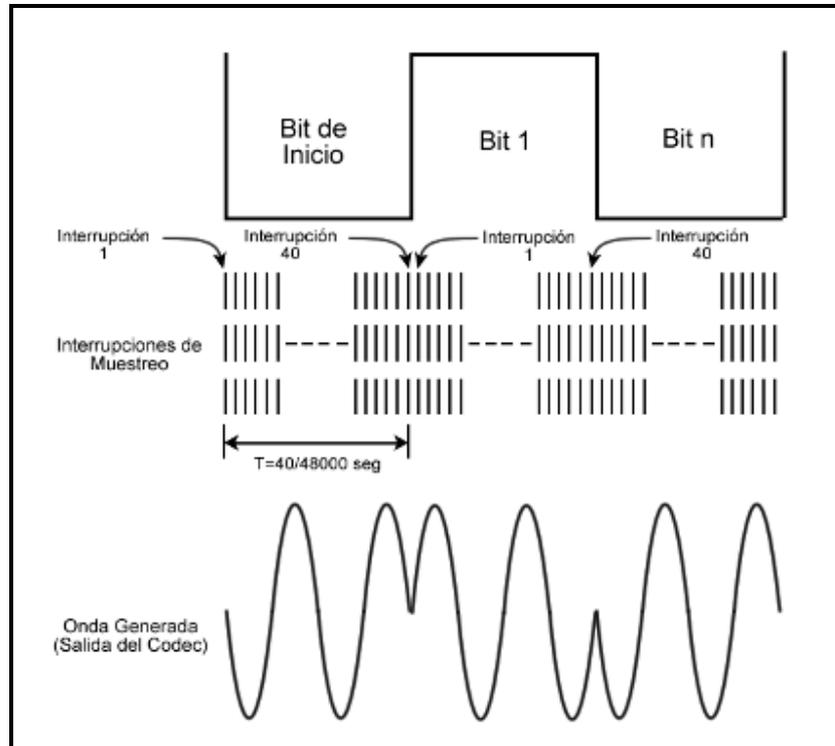
Así entonces se debe generar o no la onda portadora según el estado lógico, durante el tiempo de bit. Para lograr este tiempo de bit se toma como base el período de muestreo, se calcula cuantos períodos de muestreo están contenidos en un tiempo de bit, con este resultado se implementa un contador descendente (*cntr\_int*), que disminuye cada interrupción por muestreo, cuando el contador expira se pasa a generar la onda respectiva según el estado lógico del siguiente bit, caso contrario se mantiene generando la onda según el estado lógico del bit actual.

La velocidad de señal deseada para nuestro caso es de 1200 baudios, por tanto:

$$cntr\_int = \frac{Fm}{Vs} = \frac{48000}{1200} = 40 \quad (3.1)$$

Este contador descendente vuelve a ser cargado a su valor máximo cada vez que se trata un nuevo bit de carácter.

El contador así implementado constituye entonces un contador de interrupciones.



**Figura 3. 12:** Proceso de modulación

La determinación del estado lógico del bit se realiza con un par de instrucciones creadas para tal propósito y que forman parte del repertorio de nemónicos del lenguaje *assembler* utilizado.

Los elementos de una tabla llamada *seno*, previamente almacenada en la memoria de datos a través de un buffer circular, permiten generar la onda portadora<sup>51</sup>, de esta manera cuando se determine que el bit es un uno lógico se pasa a generar la onda portadora, caso contrario se genera una señal continua de valor cero, esto último se logra leyendo continuamente el primer valor de la tabla que por defecto es cero, por ello se hace que  $m7 = 0$  con lo cual el buffer circular *seno* no actualiza su puntero a la siguiente posición.

<sup>51</sup> Para mayor detalle puede consultarse la página 32 del presente proyecto de titulación.

Los elementos de la tabla *seno* que generará la onda portadora se los puede obtener de MatLab, el número necesario se lo puede calcular con la expresión  $1.1^{52}$

Esto explica el valor de veinte en la declaración del buffer circular que contiene la tabla con las muestras, ya que la frecuencia de muestreo y portadora son 48000 y 2400 Hz respectivamente; este buffer circular se lo inicializa a través de un archivo llamado *sin.dat* que contiene las muestras en el formato adecuado (1.15 hexadecimal).

Cuando se ha terminado de procesar un carácter completo se incrementa un contador ascendente, llamado *contador\_caracter\_tx*, se determina entonces cuantos octetos asíncronos restan por modular y transmitirse, para lo cual se usa la variable *num\_caract\_rx* que, como se vio anteriormente, guarda el número total de caracteres almacenados incluyendo bytes de entrenamiento e identificación del móvil. Antes de realizar la comparación entre estas dos variables, a la última se le resta ocho bytes, ya que estos representan los caracteres finales de la trama *GPRMC* original almacenada y que no se transmitirán, por cuanto serán reemplazados por otros que implementan el checksum calculado por un módulo anterior y símbolos de fin de trama, lo cual será analizado más adelante.

Cuando la comparación entre las variables anteriores resulte ser diferente de cero se continua con el proceso normal bit por bit y caracter a caracter, para lo cual se reinician variables a sus valores máximos y se mueven banderas según corresponda, tal como lo señala el diagrama de flujo. Si la comparación es cero la transmisión de los caracteres planificados a enviar se ha completado con éxito y se procede a modular y transmitir los caracteres finales de la trama, el ASCII \* que señala el inicio del campo *checksum*, el nuevo checksum calculado por la primera parte del módulo *edición de trama almacenada* que consta de dos caracteres, el ASCII < que señala el fin de trama y finalmente el carrier return y line feed, todo esto en reemplazo de los caracteres señalados en la figura 3.8.

El procedimiento descrito a continuación constituye la segunda parte del módulo de edición de trama, cuyo diagrama de flujo se muestra a continuación:

---

<sup>52</sup> Página 38 del presente proyecto de titulación

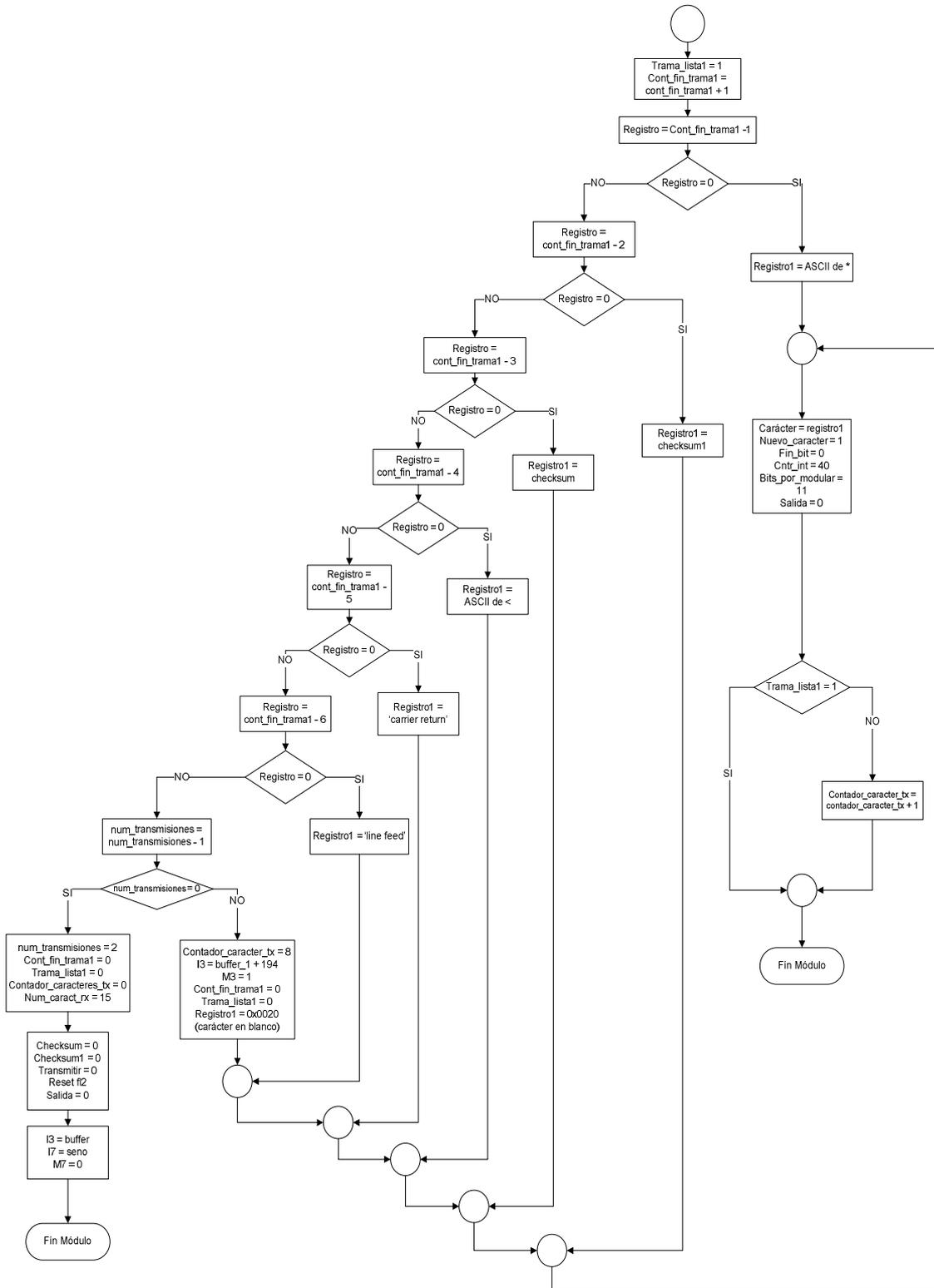


Figura 3. 13: Diagrama de flujo inserción caracteres final trama

Dicho procedimiento empieza con la variable *cont\_fin\_trama1* puesta en cero, cuando se va a modular y transmitir uno de estos seis caracteres, esta variable aumenta en uno de manera que haciendo uso de una operación matemática se determina si enviar un carácter actual o un próximo; por ejemplo, como se ha dicho, en un inicio *cont\_fin\_trama1* es cero, al entrar por primera vez, esta variable aumenta a uno, se verifica si ya se ha enviado el primer carácter restando de *cont\_fin\_trama1* el valor de uno, el resultado es cero con lo cual se pasa a transmitir el ASCII \*, la próxima vez que se entre a este módulo *cont\_fin\_trama1* valdrá dos, se comprobará si se envió el asterisco restándole de la variable la cantidad uno, como el resultado no es cero se pasa a verificar si se ha enviado el siguiente carácter restando de la misma variable la cantidad dos, ahora el resultado es cero y se procede a transmitir el primer carácter del checksum y así sucesivamente hasta completar los seis bytes finales de trama, con lo cual se ha modulado y transmitido una trama completa con todos los elementos descritos en la figura 2.2<sup>53</sup>

#### **3.1.1.5.2. Modulación FSK**

Brevemente, la modulación digital FSK hace corresponder a cada estado lógico binario una frecuencia de portadora diferente.

A continuación se presenta el diagrama de flujo y su explicación correspondiente:

---

<sup>53</sup> Página 75 del presente proyecto de titulación

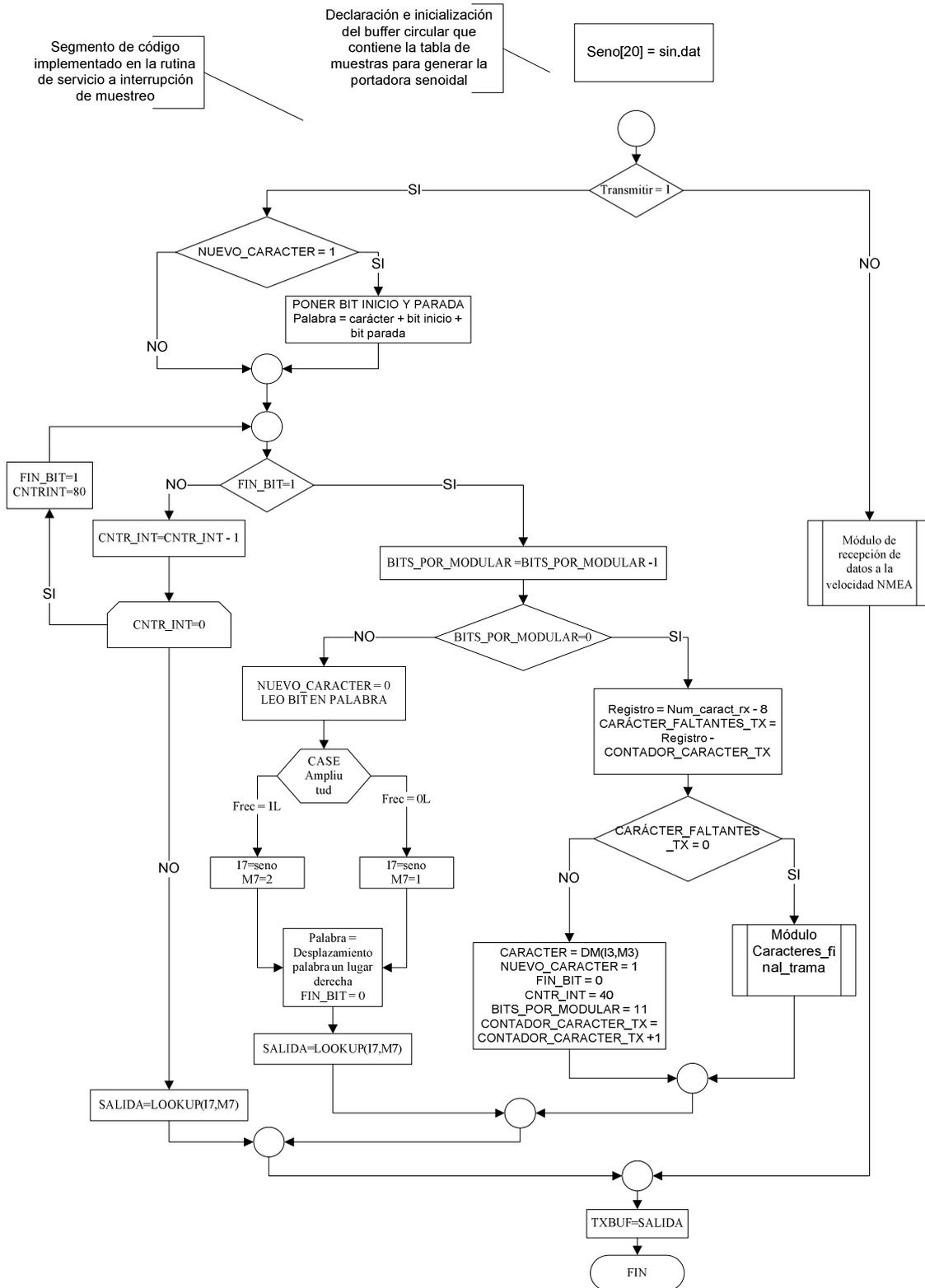


Figura 3. 14: Diagrama de flujo modulación FSK

El algoritmo implementado es básicamente el mismo que para la modulación ASK con las siguientes diferencias:

- El número de muestras necesarias para generar la onda portadora no es veinte sino cuarenta, ya que se necesita generar la onda de frecuencia de espacio que es de 1200 Hz, el cálculo correspondiente puede ser comprobado con la expresión 1.1<sup>54</sup>.
- Para generar dos ondas diferentes con una misma tabla se debe manejar los registros de modificación (M) de los punteros de los DAGs, los cuales indican cuantas localidades en el buffer circular se salta para apuntar a la siguiente, así para generar la frecuencia de espacio se usan las 40 muestras disponibles, por tanto se fija el valor de M en uno, para la frecuencia de marca se necesitan solamente 20 muestras, por tanto M se fija en dos, esto se realiza en la parte en que se ha determinado ya el estado lógico del bit en cuestión y se pasa a modularlo.

#### **3.1.1.5.3. Modulación BPSK**

Como se describió en el capítulo dos, la modulación PSK a cada estado lógico binario hace corresponder una fase de la misma portadora (igual amplitud y frecuencia), para el caso particular de BPSK la fase de cero grados es asignada al uno lógico y la de ciento ochenta grados al cero lógico.

A continuación se describe el diagrama de flujo respectivo y su correspondiente explicación:

---

<sup>54</sup> Página 38 del presente proyecto de titulación

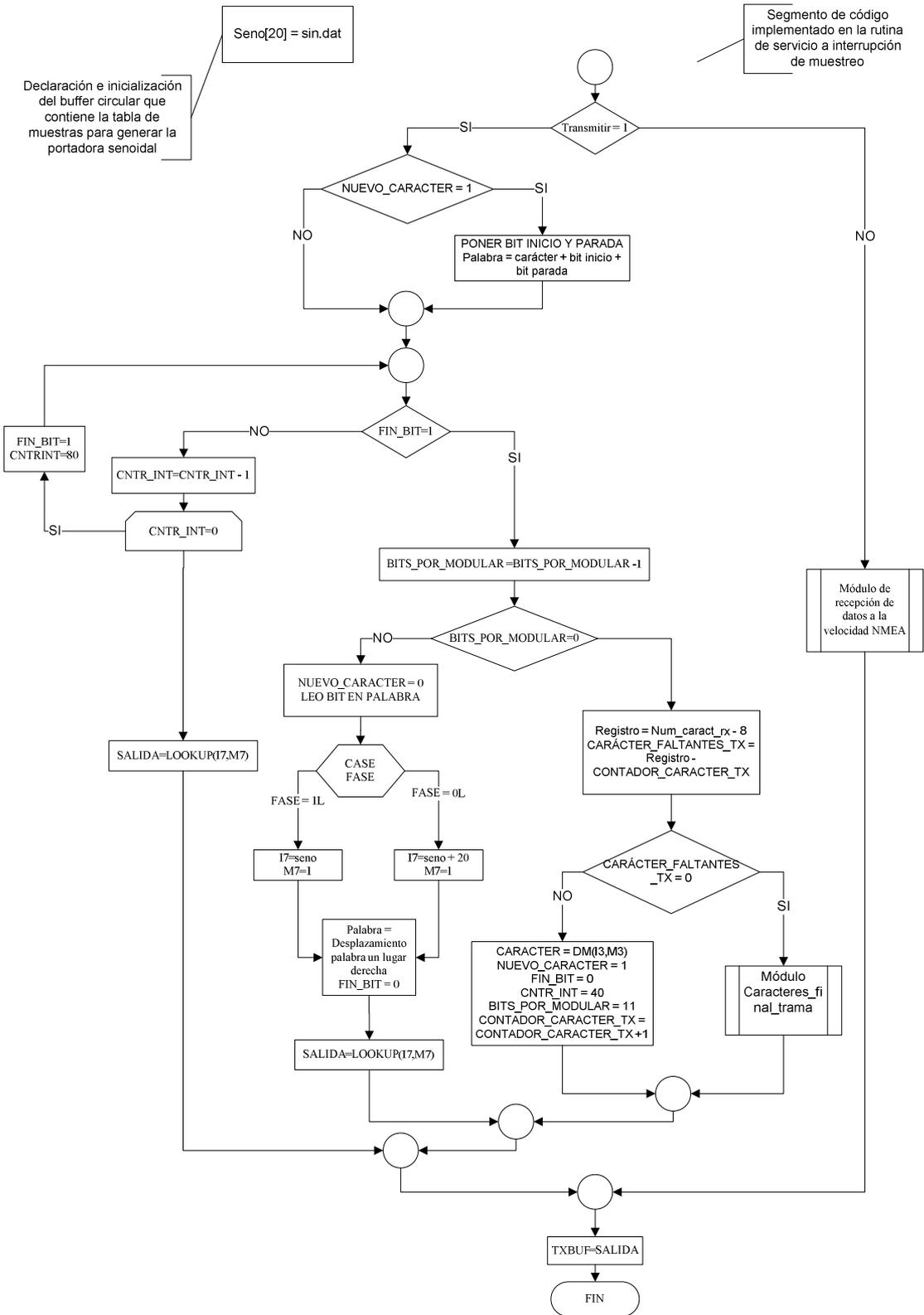


Figura 3. 15: Diagrama de flujo modulación BPSK

### 3.1.2. RECEPTOR

De idéntica forma a la realizada para el transmisor, se presenta a continuación un esquema más elaborado, que el presentado en la figura 3.1, de los procesos de firmware realizados por el receptor:

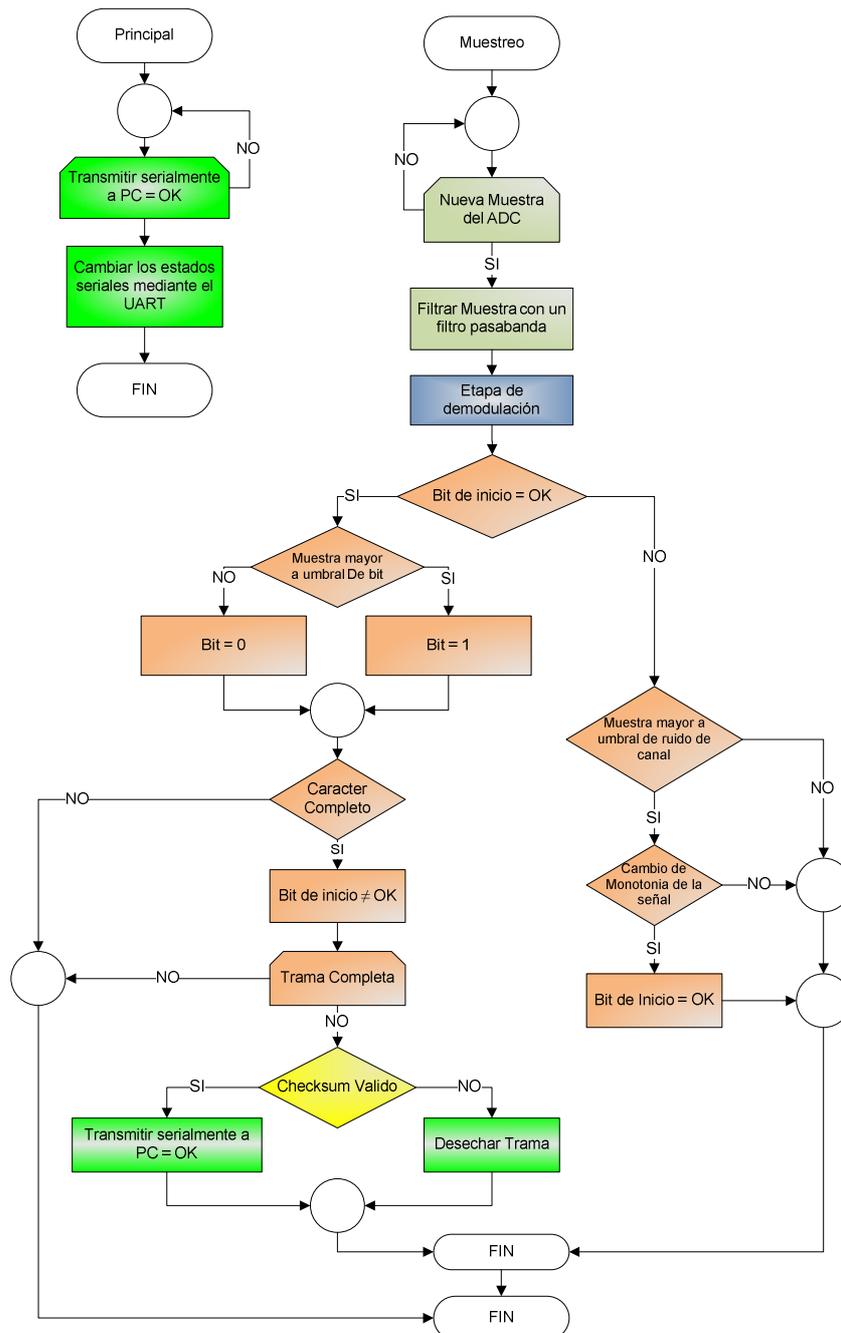
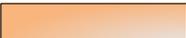


Figura 3. 16: Diagrama de flujo general de recepción

	Módulo de filtrado
	Módulo de demodulación
	Módulo de discriminación de bits
	Módulo de validación de la información recibida
	Módulo de transmisión serial

Como puede notarse en la figura 3.16 los módulos que conforman el procesamiento digital de señal en el transmisor se implementan tanto en el programa *principal* como en la rutina de servicio a la interrupción de *muestreo*, cuales son las que se implementan en una sección en particular no es una decisión arbitraria, las que se encuentran en *muestreo* necesitan de una base de tiempo fija para poder operar, dicha base temporal constituye el período de muestreo.

A continuación se pormenoriza cada uno de los módulos que realizan el procesamiento digital de señal en el receptor, se describen las etapas de modo que vayan en un orden lógico ascendente, es decir desde la captura de la forma de onda modulada hasta la transmisión serial de la información demodulada hacia el computador, primeramente se ubica un diagrama de flujo con el procedimiento detallado y posteriormente se realiza una descripción del algoritmo implementado que trata de mejorar el entendimiento de dicho diagrama de flujo y el proceso que representa.

### **3.1.2.1. Módulo de filtrado**

Como se especifica en el numeral 2.2.2.1<sup>55</sup> esta sección implementará un filtro pasa banda con el fin de limitar el espectro de la señal entrante al estrictamente necesario para la demodulación.

---

<sup>55</sup> Página 100 del presente proyecto de titulación

El algoritmo que implementa los filtros digitales FIR en tiempo real es independiente de la finalidad del filtro, sea este pasa banda, pasa bajos o pasa altos, su diagrama de flujo es el siguiente:

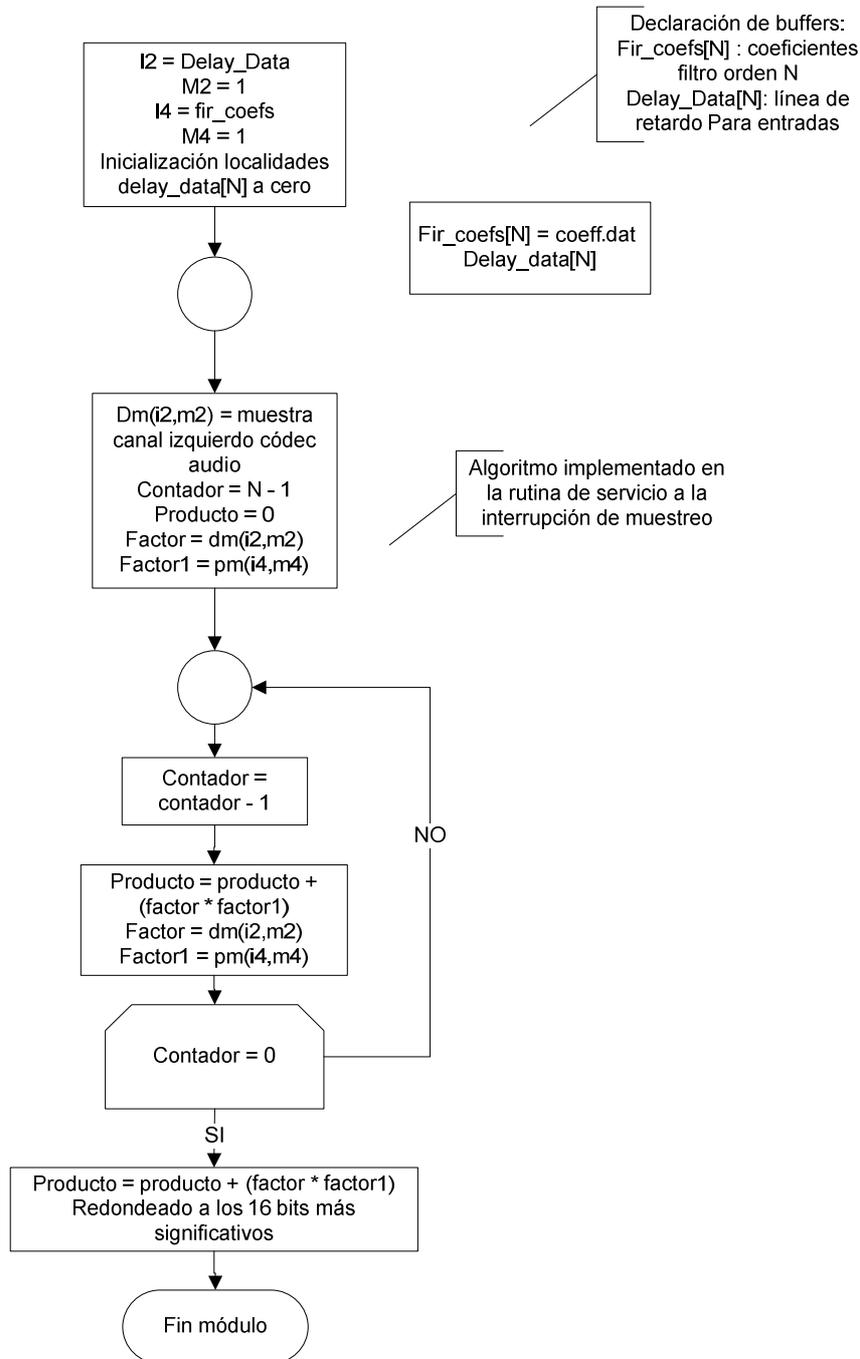


Figura 3. 17: Diagrama de flujo filtro digital FIR tiempo Real

Para implementar el filtro en tiempo real primeramente se deben obtener los coeficientes del mismo, estos determinan la naturaleza de dicho filtro, y se obtuvieron del software computacional *MatLab* tal y como se lo describió en el procedimiento explicado en el numeral 1.3.2.2.1<sup>56</sup> del capítulo primero.

Un filtro FIR debe implementar la siguiente ecuación:

$$y(n) = h(0) * x(n) + h(1) * x(n - 1) + \dots + h(N - 1) * x(n - N + 1) \quad (3.2)$$

Es decir:

$$y(n) = \sum_{k=0}^{N-1} h(k) * x(n - k) \quad (3.3)$$

Donde:

N = Orden del filtro FIR

Para cumplir este cometido se deben destinar dos buffers circulares, el primero contiene los coeficientes del filtro y se lo efectúa en memoria de programa, la cual contiene veinte y cuatro bits por localidad lo que brinda una mayor precisión; el segundo buffer circular es la denominada línea de retardo que guarda la entrada actual y las entradas anteriores, de las cuales como sabemos, depende la realización de un filtro digital FIR.

El número de localidades para cada buffer es igual al orden del filtro digital FIR.

El algoritmo implementado y descrito por el diagrama de flujo anterior, se basa en reiteradas operaciones de multiplicación y acumulación, es aquí donde la arquitectura del Procesador digital de señal ADSP – 2181 constituye una ventaja, ya que realiza estas operaciones en un solo ciclo de máquina; las N - 1 operaciones son realizadas con una precisión de 32 bits, mientras que la última se realiza con una de 16 bits redondeando el resultado final.

---

<sup>56</sup> Página 32 del presente proyecto de titulación

### 3.1.2.2. Módulo de demodulación

Como se destacó en el capítulo segundo, el proceso de demodulación es por lo general más complejo que el de modulación, esto se ve claramente reflejado en la descripción de implementación de los diferentes esquemas realizada a lo largo del numeral 3.1.1.5<sup>57</sup>, donde las diferencias entre uno y otro radican esencialmente en el manejo de los parámetros de un buffer circular para implementar la(s) onda(s) portadora(s).

Como puede visualizarse en la figura 3.16 el módulo de demodulación de la información recibida se implementa por completo en la rutina de servicio a la interrupción de *muestreo*, esto se lo realiza por facilidad, por cuanto las demás estructuras de demodulación, por ejemplo el discriminador de bits, se implementan en esta sección por necesitar de una base de tiempo fija a la cual referirse.

A continuación se describe la parte de demodulación de cada uno de los esquemas detallados en el numeral anterior, para de esta forma poder sacar conclusiones que den origen al tipo de modulación definitivo a usar, se incluye además una sección de justificación matemática que antecede al diagrama de flujo correspondiente de manera que se entienda completamente lo realizado en el mismo.

#### 3.1.2.2.1. Demodulación ASK

Justificación Matemática:

Como sabemos de la figura 2.8<sup>58</sup> la señal de entrada al demodulador puede ser  $+\text{sen}(w_c t)$  ó  $0$ . Entonces:

Entrada al demodulador:

$$S_{ASK}(t) = A * d(t) \text{sen}(w_c t); \quad d(t) = 0 \quad \text{ó} \quad d(t) = 1 \quad (3.4)$$

<sup>57</sup> Página 121 del presente proyecto de titulación

<sup>58</sup> Página 86 del presente proyecto de titulación

Aplicando valor absoluto:

$$S(t) = |A| * |d(t)| |\text{sen}(w_c t)| \quad (3.5)$$

Reemplazando:

$$S(t) = |A| * |d(t)| \sqrt{\text{sen}^2(w_c t)} \quad (3.6)$$

Elevando al cuadrado:

$$S^2(t) = A^2 * d^2(t) * \left( \frac{1}{2} - \frac{\cos(2w_c t)}{2} \right) \quad (3.7)$$

Es decir:

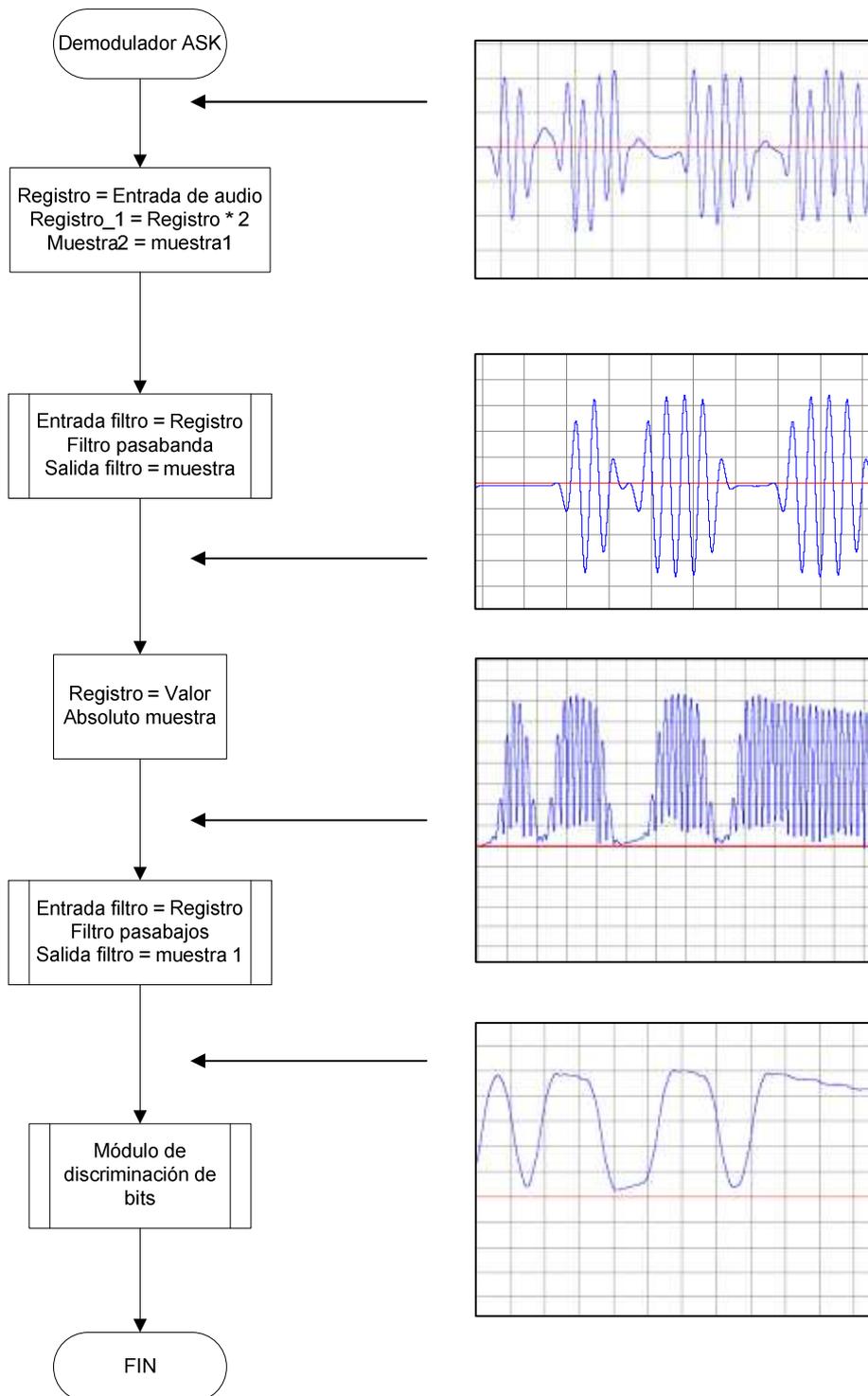
$$S_1(t) = \frac{A^2 d^2(t)}{2} - \frac{A^2 d^2(t) \cos(2w_c t)}{2} \quad (3.8)$$

Finalmente se hace pasar la señal por un filtro pasa bajos que tiene un punto de corte mucho más bajo que  $2w_c$  por lo que se tiene:

$$S_1(t) = \frac{A^2 d^2(t)}{2} \quad (3.9)$$

Si  $d(t) = 0$   $S_1(t) = 0$  correspondiente a un cero lógico, si  $d(t) = 1$   $S_1(t) = A^2/2$  correspondiente a un uno lógico.

A continuación se presenta el diagrama de flujo que implementa la demodulación ASK y su respectiva explicación:



**Figura 3. 18:** Diagrama de flujo demodulador ASK

Como puede visualizarse, al menos en estructura, el demodulador ASK es sencillo, depende básicamente de un elemento de procesamiento digital de señal, los filtros digitales FIR en tiempo real, al final de la última etapa de filtrado se dispone de la señal cuasi binaria, de la cual habrá que discriminar los bits de datos, el módulo que hace posible esta última operación se lo describe más adelante en el presente capítulo.

Al inicio del procesamiento existe un bloque de ganancia por dos, esto se realiza para compensar una atenuación a la mitad de la señal que se produce en la entrada del códec de la tarjeta de desarrollo.

Dos tipos diferentes de filtros son necesarios, uno pasa banda y por último uno pasa bajos, el primero de ellos con el objetivo de dejar pasar solamente el ancho de banda necesario y suficiente para poder recuperar la información modulada, lo cual es realizado por el módulo anteriormente descrito, y el segundo con el fin de usarlo como un detector de envolvente, todo esto se puede verificar visualmente en la figura anterior, nótese que entre los filtros pasa banda y pasa bajo existe una etapa de valor absoluto, la cual se la efectúa para obtener la parte de la señal de interés que constituye lo positivo de la forma de onda correspondiente.

El filtro pasa bajos se lo realiza mediante el mismo algoritmo de procesamiento descrito por el diagrama de flujo de la figura 3.17, lo que varía son los coeficientes del filtro, los cuales ahora determinan una respuesta diferente a la del pasa banda.

#### 3.1.2.2.2. Demodulación FSK

Como sabemos de la figura 2.11<sup>59</sup> la señal de entrada al demodulador puede ser  $+sen(w_{c1}t)$  ó  $+sen(w_{c2}t)$  . Entonces:

Entrada al demodulador:

$$S_{FSK}(t) = A * d(t)sen(w_{c1}t) + A * d(t)sen(w_{c2}t); \quad d(t) = 0 \quad \text{ó} \quad d(t) = 1 \quad (3.10)$$

---

<sup>59</sup> Página 88 del presente proyecto de titulación

Se hace pasar la señal por un filtro pasa banda que elimine la frecuencia  $w_{c2}$  de tal modo que se tiene:

$$S(t) = A * d(t) \text{sen}(w_{c1}t) \quad (3.11)$$

Aplicando valor absoluto:

$$S(t) = |A| * |d(t)| |\text{sen}(w_c t)| \quad (3.12)$$

Reemplazando:

$$S(t) = |A| * |d(t)| \sqrt{\text{sen}^2(w_c t)} \quad (3.13)$$

Elevando al cuadrado:

$$S^2(t) = A^2 * d^2(t) * \left( \frac{1}{2} - \frac{\cos(2w_c t)}{2} \right) \quad (3.14)$$

Es decir:

$$S_1(t) = \frac{A^2 d^2(t)}{2} - \frac{A^2 d^2(t) \cos(2w_c t)}{2} \quad (3.15)$$

Finalmente se hace pasar la señal por un filtro pasa bajos que tiene un punto de corte mucho más bajo que  $2w_c$  por lo que se tiene:

$$S_1(t) = \frac{A^2 d^2(t)}{2} \quad (3.16)$$

Si  $d(t) = 0$   $S_1(t) = 0$  correspondiente a un cero lógico, si  $d(t) = 1$   $S_1(t) = A^2/2$  correspondiente a un uno lógico.

A continuación se presenta el diagrama de flujo que implementa la demodulación FSK y su respectiva explicación:

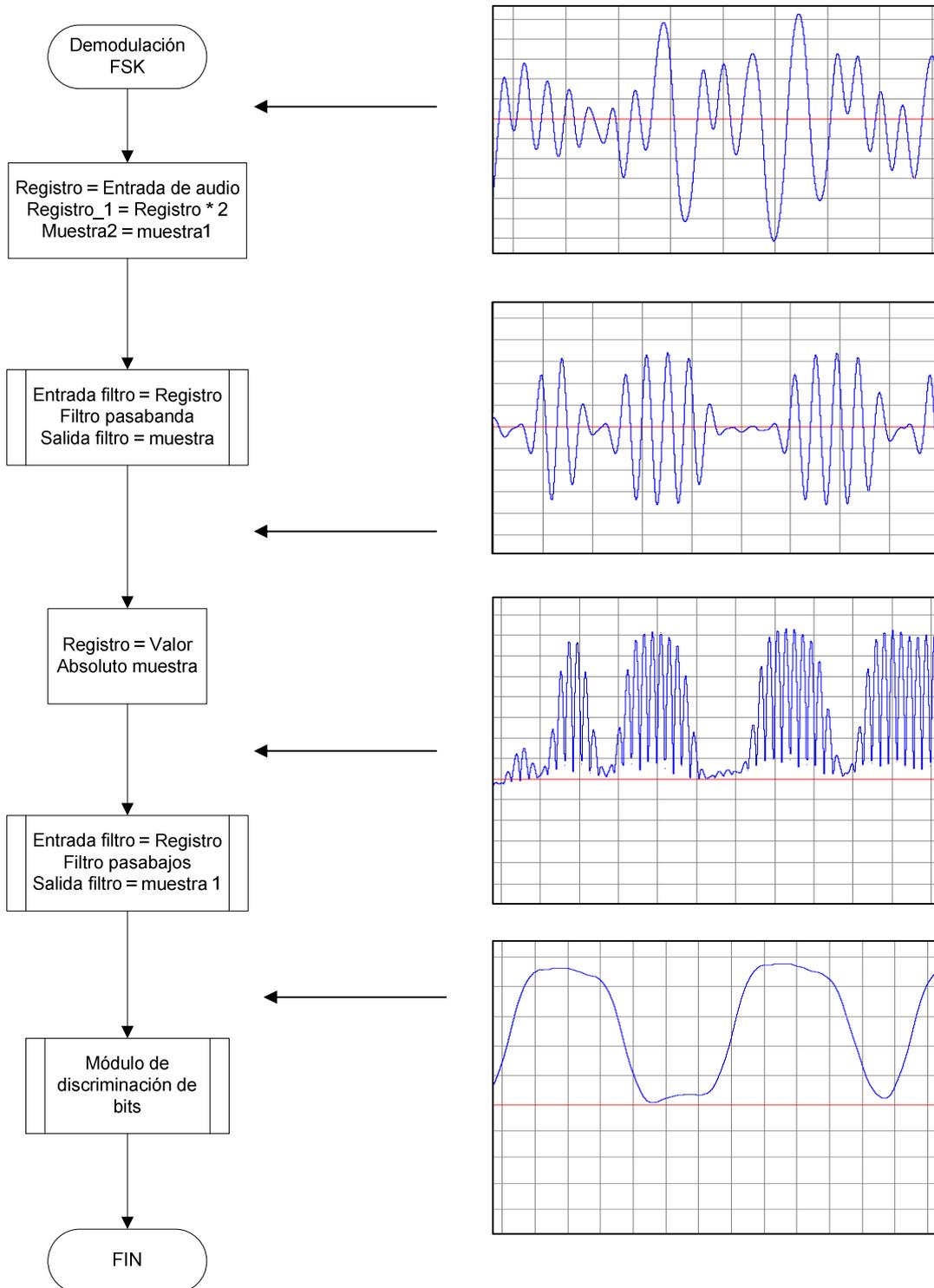


Figura 3. 19: Diagrama de flujo demodulación FSK

Los dos procesos de demodulación ASK y FSK son idénticos, el filtro pasa banda deja pasar la frecuencia de marca, con lo que la forma de onda queda en las mismas circunstancias que en ASK, por lo que el procedimiento desde este punto en adelante es idéntico que para la demodulación ASK, se ha implementado entonces una demodulación no coherente para ambos casos.

### 3.1.2.2.3. Demodulación BPSK

Justificación Matemática:

Como sabemos de la figura 2.14<sup>60</sup> la señal de entrada al demodulador puede ser  $+sen(w_c t)$  ó  $-sen(w_c t)$ . El algoritmo de demodulación PSK genera y sincroniza una reinserción de portadora que es coherente, tanto en frecuencia como en fase, con la portadora del transmisor. El proceso de reinserción se expresa como una multiplicación, lo cual se explica a continuación:

Entrada al demodulador:

$$S_{PSK}(t) = +A sen(w_c t) \quad \text{ó} \quad S_{PSK}(t) = -A sen(w_c t) \quad (3.17)$$

Luego de la reinserción de portadora:

$$S(t) = +A^2 sen^2(w_c t) \quad \text{ó} \quad S(t) = -A^2 sen^2(w_c t) \quad (3.18)$$

Después de hacer reemplazos trigonométricos resulta:

$$S(t) = \frac{A^2}{2} - \frac{A^2}{2} \cos(2w_c t) \quad \text{ó} \quad S(t) = -\frac{A^2}{2} + \frac{A^2}{2} \cos(2w_c t) \quad (3.19)$$

Finalmente se hace pasar las señales por un filtro pasa bajos que tiene un punto de corte mucho más bajo que  $2w_c$  por lo que se tiene:

$$S(t) = \frac{A^2}{2} \text{ (Para 1 lógico)} \quad \text{y} \quad S(t) = -\frac{A^2}{2} \text{ (Para 0 lógico)} \quad (3.20)$$

A continuación se presenta el diagrama de flujo que implementa la demodulación BPSK y su respectiva explicación:

---

<sup>60</sup> Página 91 del presente proyecto de titulación

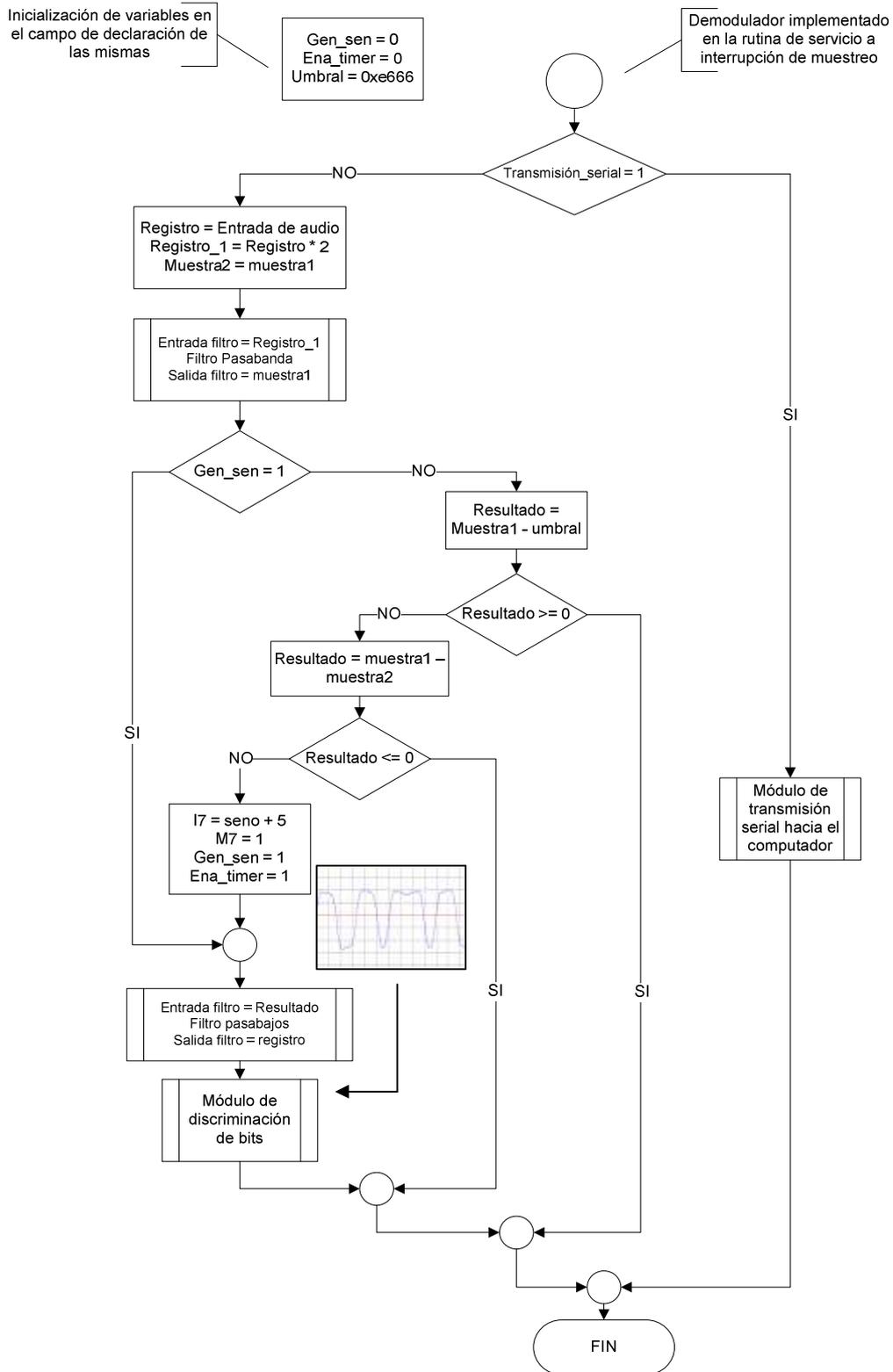


Figura 3. 20: Diagrama de flujo demodulación PSK

Este procedimiento es el más complejo entre todos los de demodulación que se ha visto hasta el momento por ser coherente, es decir hay una reinserción de portadora que lleva nuevamente la información modulada a banda base, de tal suerte que la señal modulante puede ser obtenida fácilmente aplicando un filtro pasa bajos.

Así entonces la señal recibida se multiplica por el valor de la onda seno con fase normal (la onda seno empieza en cero grados), previo a este paso se debe sincronizar la señal recibida con dicha onda seno. Se busca un cambio de monotonía (punto mínimo) en la señal recibida<sup>61</sup>, una vez hallado, este valor se lo multiplica con su correspondiente en la onda seno (valor en la tabla que representa 90 grados), de esta forma se inicia el proceso sincronizado de reinserción de portadora, dicho procedimiento se repite muestra a muestra con cada carácter, y finaliza cuando se produce una interrupción del timer la cual es programada para efectuarse cuando se ha sincronizado la multiplicación de la portadora seno con la señal correspondiente a un carácter completo.

Se prevee que la señal proveniente del ambiente inalámbrico puede estar afectada en amplitud debido a irregularidades, de esta manera primeramente se busca llegar a un determinado umbral de señal por debajo del cual se empieza a buscar el cambio de monotonía en la señal entrante.

La reinserción de portadora produce el efecto de traslado de frecuencias hacia la banda base, de manera que después de realizado este procedimiento coherente lo que resta por hacer es aplicar un filtro pasa bajos y comenzar con el proceso de discriminación explicado posteriormente.

Este proceso se suspende si se ha dado la orden de transmitir serialmente la información recibida hacia el computador, a través del seteo de la bandera *transmisión\_serial*.

---

<sup>61</sup> Se busca un punto mínimo porque la parte de la señal analizada corresponde a un cero lógico.

Antes de iniciarse el proceso de sincronización y reinserción, la señal recibida se multiplica por dos, ya que como se mencionó anteriormente la entrada de audio posee un divisor de voltaje para dos, de esta manera se compensa esta atenuación y además se aplica un filtro digital FIR pasa banda con el objetivo de limitar el espectro de la señal recibida a lo absolutamente necesario para poder recuperar la información, lo cual es realizado por el módulo de *filtrado*.

El proceso finaliza como siempre entregando el control a la etapa de discriminación de bits.

### **3.1.2.3. Módulo de discriminación de bits**

Esta etapa es parte del procedimiento de demodulación y evidentemente es común a los tres esquemas explicados anteriormente, salvo con ciertas particularidades para cada proceso evaluatorio de cuasi bits que serán señaladas más adelante, el módulo en discusión provee además de un nivel de inteligencia a la demodulación como se verá a continuación.

Se presenta entonces el diagrama de flujo respectivo y su explicación:

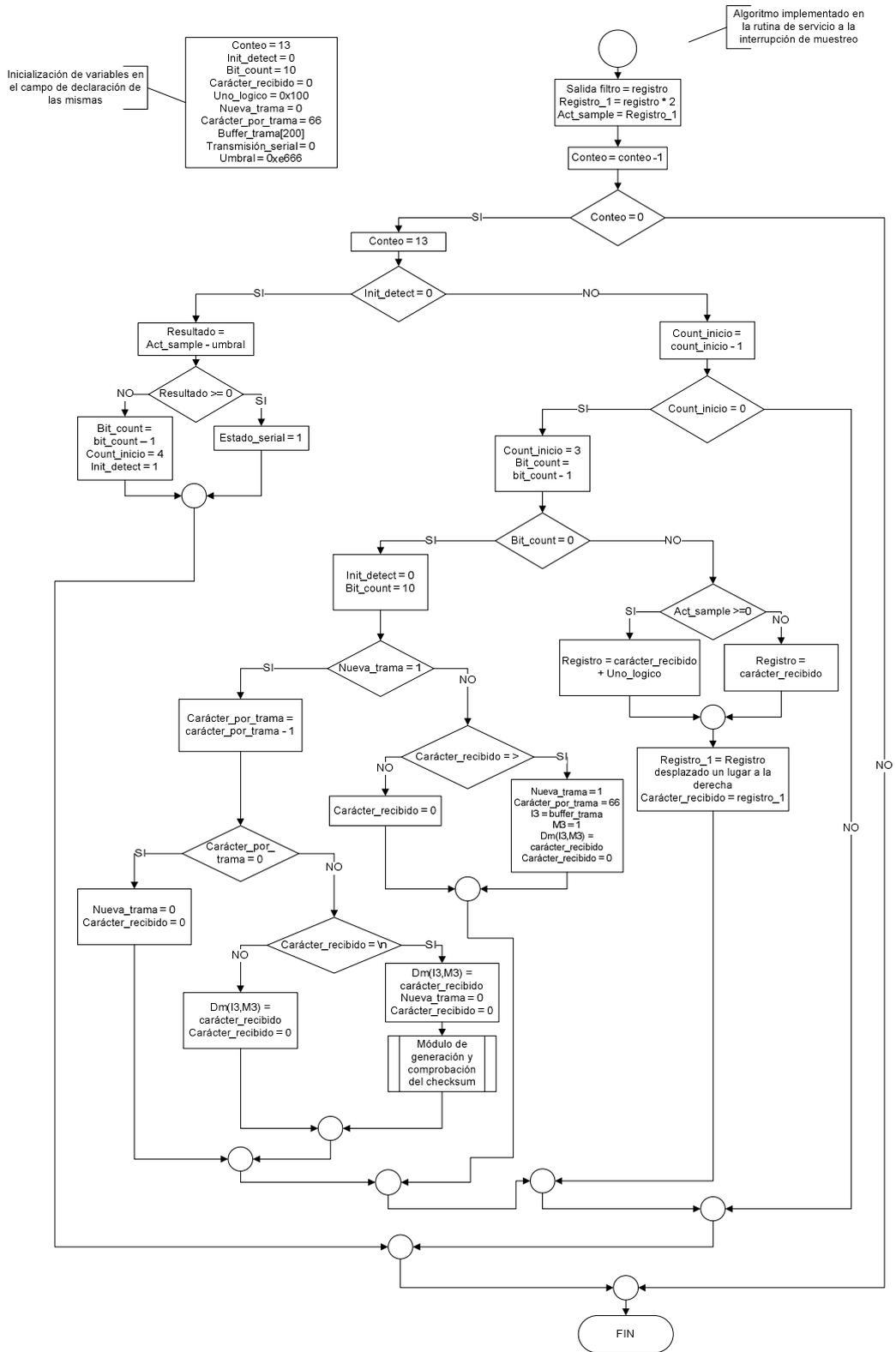
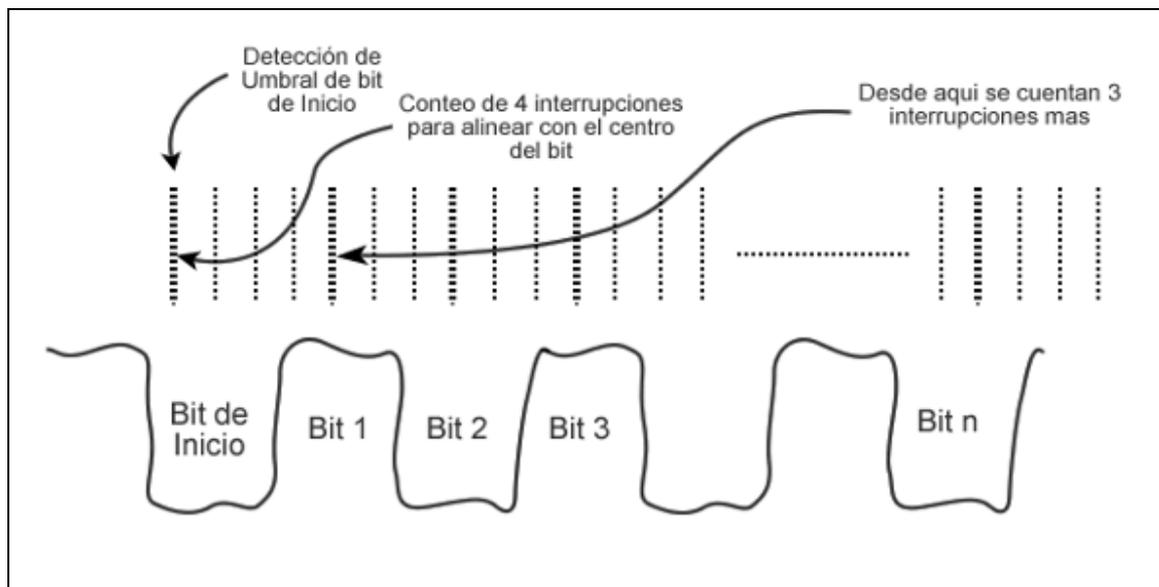


Figura 3. 21: Diagrama de flujo discriminación bits PSK

Se describirá a continuación el procedimiento efectuado para el esquema PSK, posteriormente se enumerarán los cambios a efectuarse para poder implementar ASK y FSK.

El proceso de discriminación se lo realiza tomando en cuenta el tiempo de símbolo, es decir la señal cuasi binaria se debe muestrear en los instantes precisos para determinar los estados lógicos respectivos, este instante debe ser aproximadamente la mitad de tiempo de bit, para esto una vez más se toma como referencia de tiempo el período de muestreo, tal como se lo hizo para generar la señal PSK a la velocidad de señal escogida, con la diferencia que el contador de interrupciones de muestreo (*conteo*) se carga con un número tres veces menor que el que se tendría para obtener la velocidad de señal, esto provee la suficiente resolución para determinar la naturaleza de un cuasi bit, ya que se ingresaría a cada uno cada tres veces, pudiendo determinar entonces el instante más adecuado para su discriminación.



**Figura 3. 22:** Proceso de discriminación de bits

El primer bit es el de inicio y no constituye un dato válido, sino el comienzo de un estado de alerta, previniendo la llegada de bits de datos; inicialmente el proceso debe reconocer el bit de inicio y dejarlo pasar para iniciar el reconocimiento de los

bits de datos. Para el esquema PSK se envió como bit de inicio un cero lógico, el algoritmo lo identifica a través del sobrepaso de un umbral (0xe666, correspondiente a -0.2 en formato 1.15), a partir de aquí se cuentan cuatro interrupciones de muestreo hasta alinearse con el centro del bit siguiente, que ya es un dato válido, a partir de entonces cada tres interrupciones de muestreo se toma una muestra para determinar el estado lógico del bit en cuestión, tal como se ilustra en la figura anterior.

La identificación del bit es sencilla, se verifica si la muestra es mayor o menor que cero haciendo corresponder a un uno o un cero lógico respectivamente; cada bit es almacenado en una variable denominada *carácter\_recibido* hasta que se completen los 8 bits de datos; si se trata del primer carácter se lo compara con el ASCII > tomado como inicio de trama, si se trata de este, se lo almacena y en adelante el proceso es buscar el fin de trama simbolizado con el ASCII \r, si se trata de este último se lo guarda y se entrega el control al módulo de *validación de datos*, de lo contrario se entiende que el fin de trama aun no ha sido alcanzado y se procede a guardar el carácter recibido en el buffer destinado para la trama; si no se trata del ASCII > se limpia la variable *carácter\_recibido* y se reinicia el proceso en busca de dicho byte.

Adicionalmente si no se halla el carácter de fin (\r) en sesenta y cinco intentos<sup>62</sup> igualmente se desecha la trama actual por considerarla viciada.

Para la etapa de discriminación de los esquemas ASK y FSK se toma como base lo explicado anteriormente tomando en consideración los siguientes puntos:

- El umbral de detección del bit de inicio para ASK y FSK es positivo, recuérdese que para PSK el bit de inicio es un cero lógico y por tanto la señal cuasi binaria correspondiente está por debajo del nivel cero (Figura 3.20) y por ello su umbral es negativo (0xe666); entonces para las primeras el umbral es 0x199A correspondiente a 0.2 en formato 1.15. El hecho de que se necesite un umbral

---

<sup>62</sup> Número máximo de caracteres enviados por el transmisor

positivo viene dado porque el bit de inicio para estos esquemas es uno lógico y además porque se aplicó necesariamente un valor absoluto a las muestras.

- Al aplicarse el valor absoluto todas las muestras se convirtieron en positivas de tal manera que ya no podrá tomarse como valor referencial el nivel cero para determinar la naturaleza del cuasi bit, se toma entonces en consideración el umbral anteriormente mencionado.

#### **3.1.2.4. Módulo de validación de la información recibida**

Esta etapa verifica la integridad de la información recibida, calculando para esto un checksum de idéntica forma a la realizada en el transmisor, dicho checksum se lo compara con el recibido, de ser iguales se ha recibido correctamente una trama caso contrario se la desecha.

El diagrama de flujo y su respectiva explicación se muestran a continuación:

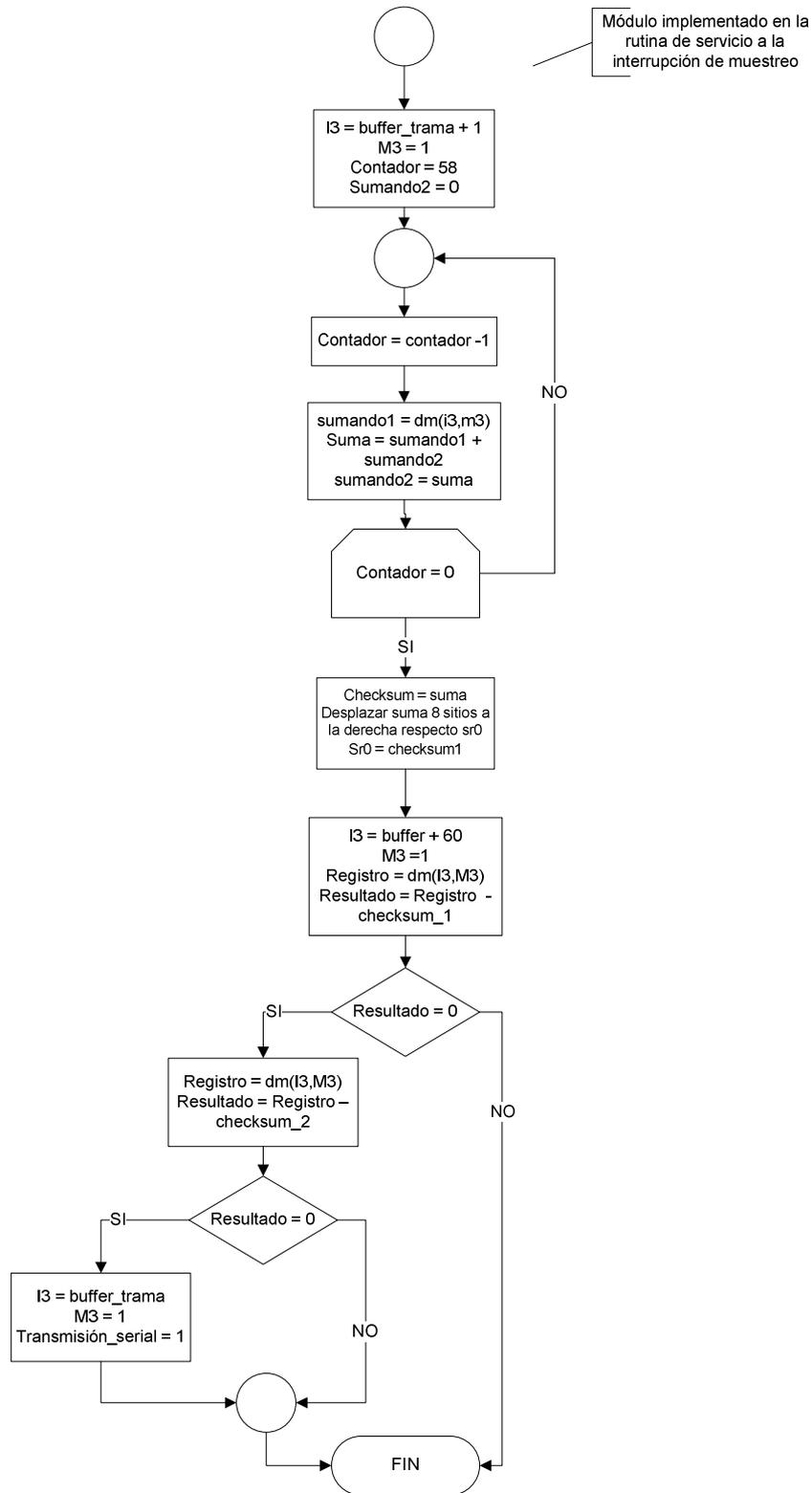


Figura 3. 23: Diagrama de flujo de validación de información recibida

Los caracteres que componen el checksum se constituyen de idéntica forma a la explicada en el numeral 3.1.1.3<sup>63</sup>, los caracteres uno y dos de dicho checksum generados en el receptor se almacenan en las variables *checksum* y *checksum1* respectivamente, entonces se compara miembro a miembro con los recibidos, tales se encuentran almacenados en las posiciones 60 y 61 respectivamente del buffer que contiene la trama. Si son idénticos ambos pares de caracteres, se valida la trama actual y se sede el control al siguiente módulo de programa encargado de enviar los datos recibidos y validados a través del puerto serial hacia el computador, de lo contrario se desecha por completo la trama analizada por considerarla errónea en alguno de sus campos.

### **3.1.2.5. Módulo de transmisión serial**

Constituye la etapa final del Firmware y esta se relaciona con el software que tomará los datos para visualizarlos en el computador.

Como puede visualizarse en la figura 3.16<sup>64</sup> este módulo se lo implementa tanto en la sección *principal* de programa, como en la rutina de servicio a la rutina de interrupción de *muestreo*.

El diagrama de flujo de la parte realizada en la sección de muestreo y su correspondiente explicación se detalla a continuación:

---

<sup>63</sup> Página 114 del presente proyecto de titulación

<sup>64</sup> Página 133 del presente proyecto de titulación

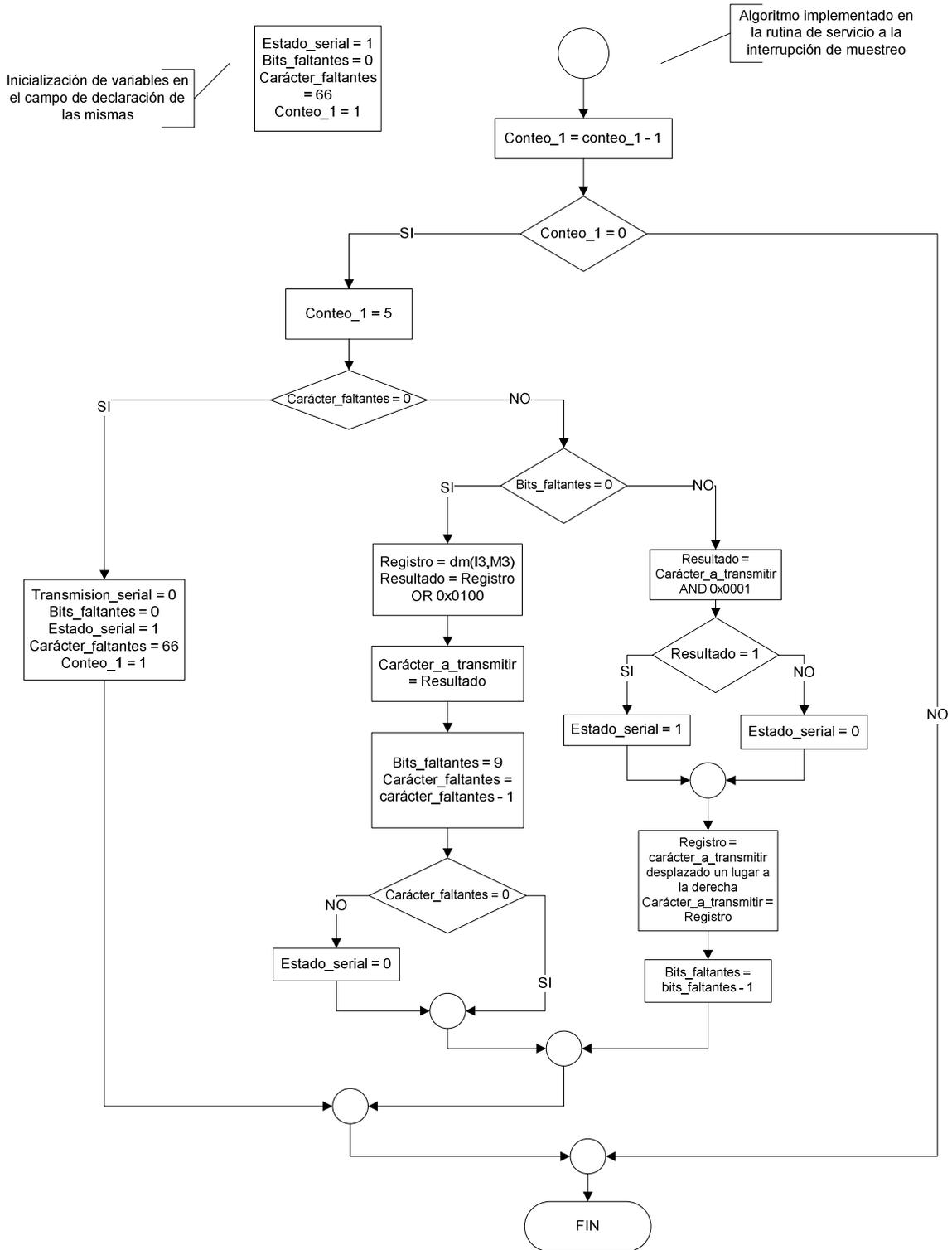


Figura 3. 24: Diagrama de flujo de transmisión serial a 9600 bps

Como se ha venido manifestando, la razón para hacer uso de la rutina de servicio a la interrupción de *muestreo* es la necesidad de una base de tiempo fija de referencia para poder implementar la velocidad de transmisión hacia el computador. Se emplea para tal objetivo un contador regresivo denominado *conteo\_1*, cuyo valor máximo se calcula con la siguiente expresión:

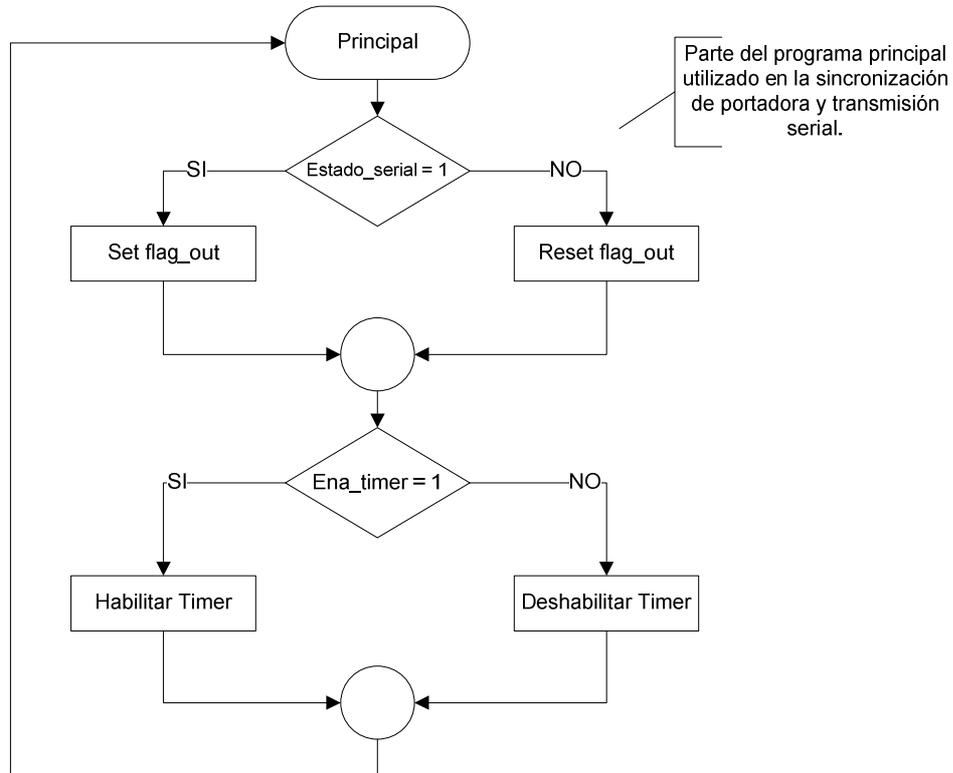
$$\text{conteo\_1} = \frac{f_m}{V_{tx}} \quad (3.3)$$

Donde:

$f_m$  = frecuencia de muestreo empleada en el sistema DSP.

$V_{tx}$  = Velocidad de transmisión deseada de los bits hacia el computador.

Para nuestro caso se usa una frecuencia de muestreo de 48000 Hz y una velocidad de transmisión escogida de 9600 bps, con lo cual *conteo\_1* resulta ser de cinco, es decir cada vez que este valor se haga cero se identificará la naturaleza del bit de carácter en cuestión, dependiendo de esto último se dispondrá un valor para la bandera *Estado\_serial*, cuya verificación será realizada por el programa principal enviando por la interfaz serial un estado alto o un bajo que identificará un uno ó un cero lógico respectivamente. La parte del programa principal mencionada es la siguiente:



**Figura 3. 25:** Chequeo de banderas para transmisión serial

La sección que hace referencia al chequeo de la bandera *Ena\_timer* pertenece a la etapa de sincronización de portadora tal y como se explicó en el numeral 3.1.2.2.3<sup>65</sup>, la cual permite iniciar y finalizar el proceso de sincronización de portadora en la demodulación BPSK.

### 3.1.3. TIPO DE MODULACIÓN DEFINITIVA A UTILIZAR

Los aspectos importantes a evaluar en la elección del esquema de modulación son:

- Ancho de banda utilizado
- Inmunidad al ruido
- Aspectos de implementación

<sup>65</sup> Página 143 del presente proyecto de titulación

En cuanto se refiere al ancho de banda utilizado por cada uno de los esquemas de modulación, se puede concluir que el que más necesita de este recurso es FSK, mientras que para ASK y PSK el ancho de banda necesitado es el mismo para una determinada velocidad de señal y numéricamente idéntico a esta.

La naturaleza misma de ASK la hace excesivamente sensible al ruido del canal, ya que la información es llevada en su amplitud. Al usar un procedimiento de demodulación no coherente para FSK se obtuvo como consecuencia una mayor sensibilidad al ruido de canal, en contraste, el esquema PSK presenta una mayor inmunidad al ruido por cuanto la fase de la portadora es la que se encuentra en función de la modulante, además de haberse utilizado un procedimiento coherente en demodulación.

En los aspectos relacionados con la implementación se puede mencionar que para FSK y ASK las señales demoduladas tienen la forma de una onda pseudo cuadrada, es decir un estado alto y un estado bajo oscilante de manera irregular en el estado bajo (Cero), por otro lado la señal demodulada PSK tiene la forma de una onda del tipo NRZ (*No Return to Zero*), es decir en ningún momento hay oscilaciones entorno al cero, esto puede ser verificado en las figuras 3.18, 3.19 y 3.20 respectivamente.

Esto tiene las siguientes implicaciones:

- En PSK se utiliza todo el rango disponible (*dado por el formato configurado y usado*) para la representación numérica de las muestras a ser procesadas en el ADSP.
- En FSK y ASK, por otro lado, se utiliza una parte del rango numérico disponible, aproximándose a la mitad de este.
- Para FSK, en el proceso de modulación, es necesario incluir un uno lógico como bit de inicio, por cuanto en la discriminación se deberá reconocer este uno, para poner de inmediato en la parte serial un estado bajo que indique la llegada de datos, en ASK y FSK no es posible modular un cero como bit de

inicio, ya que no se puede reconocer este, cuando el estado de reposo de la señal es en nivel cero.

- En PSK, dada la similaridad con una señal NRZ, es posible detectar un cero, para de esta forma poner de manera inmediata un estado bajo en la parte serial e indicar el comienzo de datos válidos.

Es evidente, además, que implementar el esquema PSK trae consigo mayor dificultad por la estructura de su demodulador, en contraposición, sus ventajas relacionadas con mayor inmunidad al ruido, ancho de banda mínimo utilizado y uso completo del rango numérico en tarjeta lo hacen más atractivo.

El siguiente cuadro recoge de manera resumida las conclusiones resultado de los análisis previos en base a los puntos establecidos en el párrafo anterior.

Esquema de modulación	Ancho de Banda Usado	Inmunidad al ruido	Facilidad Implementación
<i>ASK</i>	<i>Igual a la velocidad de señal escogida</i>	<i>Susceptible</i>	<i>Fácil</i>
<i>FSK</i>	<i>Mayor a la velocidad de señal escogida</i>	<i>Susceptible para demodulación no coherente</i>	<i>Fácil</i>
<i>BPSK</i>	<i>Igual a la Velocidad de Señal</i>	<i>No susceptible</i>	<i>Más compleja que las anteriores</i>

**Tabla 3. 1:** Comparación de esquemas de modulación

Si bien el esquema BPSK presenta una mayor dificultad en la implementación, en lo que se refiere al demodulador, presenta una mayor eficiencia en el uso de recursos, además de presentar una mayor inmunidad al ruido que los otros esquemas, de tal forma que será el que se adopte para implementar el radio módem que forma parte del sistema LAV, se dará origen entonces al prototipo final con el cual se realizarán las respectivas pruebas de funcionamiento finales.

### 3.1.4. IMPLEMENTACIÓN FINAL DEL FIRMWARE

Todo el desarrollo de cada uno de los módulos fue realizado utilizando el programa *Host* de la tarjeta de desarrollo EZ Kit Lite ADSP 2181 llamado *monitor*, el cual

ejecuta el programa cargado serialmente como una subrutina del mismo, tal y como se mencionó en el numeral 1.4.4<sup>66</sup> del capítulo primero, dicho programa se mantiene en memoria mientras exista alimentación de energía eléctrica, este es entonces un problema ya que se requiere que el programa final se ejecute inmediatamente después de alimentado el sistema y sin necesidad de hacer la carga serial del programa cada vez.

Este problema se soluciona almacenando el programa final en memorias UVPR0M compatibles con la tarjeta de desarrollo y que reemplazan a la EPROM original que contiene el programa *monitor*.

El kit de desarrollo trae consigo entre sus herramientas una apropiada para generar el archivo que se grabará en las memorias UVPR0M, se trata de un ejecutable al cual se le entregan ciertos parámetros para que a la salida se obtenga dicho archivo.

La sintaxis empleada es la siguiente:

```
Spl21 nombre_archivo prom_nombre -loader -2181 -i
```

Donde:

Spl21 Es la herramienta ejecutable

Nombre\_archivo Es el nombre del .exe producto de la compilación del VisualDSP

Prom\_nombre Es el nombre deseado del archivo de salida

-loader Es el parámetro que permite la inclusión del código que efectúa la descarga del programa de la UVPR0M hacia la memoria interna del DSP, es lo primero que se ejecuta luego de alimentado el sistema.

-2181 Crea el código de máquina específico para el DSP 2181

-i Especifica el formato hexadecimal en este caso INTEL

---

<sup>66</sup> Página 63 del presente proyecto de titulación



**Figura 3. 26:** Memorias UVPROM Utilizadas

## **3.2. SOFTWARE**

Como se mencionó anteriormente es la etapa que se relaciona con el firmware y con el usuario que vigila el movimiento de los activos móviles, dicho de esta manera, la parte de software puede ser dividida en tres partes, la primera encargada de procesar la información enviada desde la fase de Firmware, la segunda delegada a graficar la ubicación del móvil en un mapa digital a través del software SIG<sup>67</sup> y la tercera destinada a registrar y almacenar el histórico posicional en una base de datos.

### **3.2.1. MÓDULO DE TRATAMIENTO DE DATOS**

La operación del módulo de datos puede resumirse simplemente a escuchar el puerto serial, capturar los datos y generar las entradas para el software SIG google Earth y la base de datos.

Dicho módulo fue gentilmente proporcionado por la empresa de soluciones tecnológicas *Watching*, el cual fue enteramente desarrollado en JAVA, un lenguaje

---

<sup>67</sup> Sistema de Información Geográfica

de tendencia actual por sus ventajas relacionadas con licencia gratuita de uso y características de portabilidad entre las más destacadas.

En JAVA, y en cualquier lenguaje orientado a objetos, existen las denominadas *clases*, las cuales marcan la estructura básica de un programa, así en este módulo se tienen cinco principales, las cuales se denominan:

- ***Mytestport.java***: Es la clase principal, llamada a ejecutarse en primer lugar luego de haberse establecido la máquina virtual JAVA, estableciendo nexos entre las clases subsiguientes.
- ***Downloaddata.java***: Su función principal es configurar, escuchar y almacenar los datos provenientes del puerto serial.
- ***Dao.java***: Encargado de abrir y cerrar la base de datos para almacenar la información de la trama que contiene la posición.
- ***TrackGPRMC.java***: Se encarga de separar por campos la información demodulada, ya que existen diversos tipos adicionales a los de posición, por ejemplo hora, fecha, velocidad, ángulo respecto del norte, etc.
- ***google.java***: Se encarga de generar el archivo de extensión *.KML*<sup>68</sup> que será utilizado por el google Earth para graficar las diferentes posiciones que adopten los móviles.

### 3.2.2. VISUALIZACIÓN DE RESULTADOS EN GOOGLE EARTH

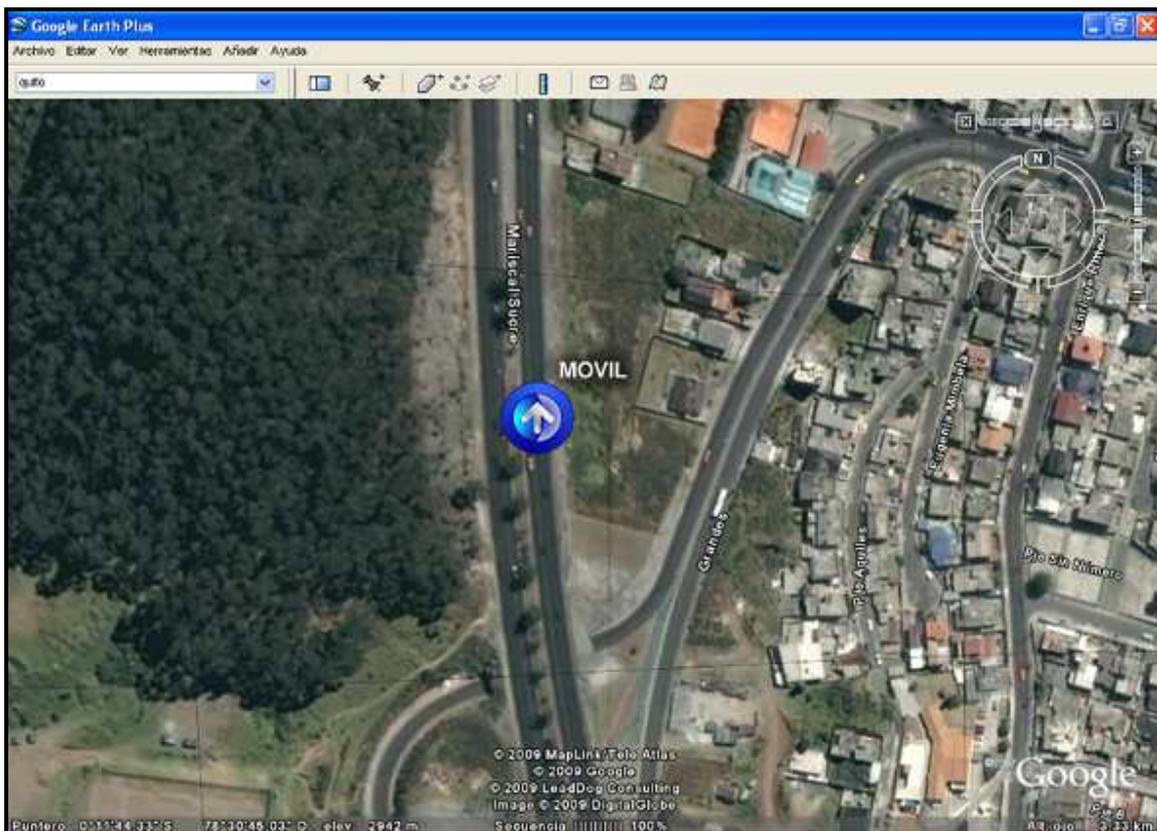
Como se mencionó en el numeral 1.2.4<sup>69</sup> del capítulo primero, el Software SIG a utilizar en el presente proyecto de titulación es el Google Earth por sus ya descritas ventajas.

---

<sup>68</sup> Acrónimo de Keyhole Markup Language, es un formato de archivo que se utiliza para mostrar información geográfica en navegadores terrestres como Google Earth y Google Maps.

<sup>69</sup> Página 18 del presente proyecto de titulación

El módulo de procesamiento de datos se encarga de generar el archivo *KML* que es utilizado por Google Earth para graficar la posición actualizada de los móviles, de esta manera cada vez que llega y se demodula una trama válida se produce una actualización en tiempo real de la posición del móvil en pantalla verificable por el usuario.



**Figura 3. 27:** Visualización de resultados en Google Earth

### 3.2.3. ALMACENAMIENTO DE INFORMACIÓN RECIBIDA EN BASE DE DATOS

La base de datos generada se encuentra en Microsoft Access, mediante esta se puede acceder al registro histórico de las posiciones ocupadas por los diferentes vehículos en intervalo de tiempo de interés, sin embargo lo almacenado no solamente tiene que ver con ubicación geográfica sino además con fecha, hora, rumbo y velocidad que tuvo el auto en tal posición.

ID	Hora	Estado	Latitud	Campo5	Longitud	Campo7	Velocidad	Angulo Res	Fecha	Checksum
DSP1	174920	A	121.197 S		78.295.293 W		000.0	170.6	200509	*BÚ
DSP1	174940	A	121.191 S		78.295.290 W		000.0	038.3	200509	*BÓ
DSP1	175000	A	121.173 S		78.295.272 W		000.0	052.5	200509	*BÁ
DSP1	175020	A	121.170 S		78.295.267 W		000.0	038.3	200509	*BÉ
DSP1	175040	A	121.172 S		78.295.266 W		000.0	094.3	200509	*BÍ
DSP1	175100	A	121.170 S		78.295.265 W		001.3	215.5	200509	*BÉ
DSP1	175120	A	121.168 S		78.295.259 W		000.0	047.0	200509	*BØ
DSP1	175140	A	121.167 S		78.295.258 W		000.0	051.5	200509	*BØ
DSP1	175200	A	121.176 S		78.295.260 W		000.0	193.2	200509	*BÉ
DSP1	175220	A	121.174 S		78.295.258 W		000.0	044.9	200509	*BÓ
DSP1	175240	A	121.153 S		78.295.243 W		000.0	036.7	200509	*BÉ
DSP1	175300	A	121.155 S		78.295.244 W		000.0	063.4	200509	*BÈ
DSP1	175320	A	121.154 S		78.295.244 W		000.0	044.5	200509	*BÉ
DSP1	175340	A	121.157 S		78.295.246 W		000.0	117.7	200509	*BÓ
DSP1	175400	A	121.153 S		78.295.244 W		000.0	039.5	200509	*BÉ
DSP1	175420	A	121.156 S		78.295.244 W		000.0	097.1	200509	*BØ
DSP1	175500	A	121.158 S		78.295.246 W		000.0	199.8	200509	*BÝ
DSP1	175520	A	121.179 S		78.295.257 W		000.0	204.9	200509	*BØ
DSP1	175540	A	121.195 S		78.295.267 W		000.0	205.9	200509	*BÚ
DSP1	175620	A	121.204 S		78.295.274 W		000.0	207.9	200509	*BØ
DSP1	175640	A	121.207 S		78.295.276 W		000.0	202.4	200509	*BÍ
DSP1	175700	A	121.212 S		78.295.280 W		000.0	209.2	200509	*BÆ

**Figura 3. 28:** Bases de Datos de localización en Access

El manejo adecuado de la base de datos obtenida, permite realizar filtros de manera que se visualice la información de interés en determinado momento, por ejemplo mostrar únicamente los móviles que excedieron el límite de velocidad urbano, o aquellos que estuvieron en determinada zona geográfica en un intervalo de tiempo, etc.

### 3.3. HARDWARE (ACOPLAMIENTO FÍSICO DE ELEMENTOS)

En los numerales 2.1.3<sup>70</sup> y 2.2.1<sup>71</sup> del capítulo segundo se hizo hincapié en la necesidad de poseer ciertos elementos de hardware que faciliten el acoplamiento físico entre los subsistemas de voz y datos.

Los numerales siguientes dentro de la presente sección hacen referencia al desarrollo de estos elementos y su implementación física.

<sup>70</sup> Página 78 del presente proyecto de titulación

<sup>71</sup> Página 97 del presente proyecto de titulación

### 3.3.1. INTERFAZ TARJETA DE DESARROLLO – RADIO BASE

Esta interfaz viabiliza la apertura automática del canal de comunicaciones para la transmisión de información de navegación y asimismo permite la convivencia del sistema de datos con el de voz, además de poseer ciertos elementos visuales de identificación de estado.

Dicha tarjeta es completamente manejada por el módulo de procesamiento digital de señal en transmisión llamado de *evaluación de apertura automática de canal*.

#### 3.3.1.1. Esquemático del circuito

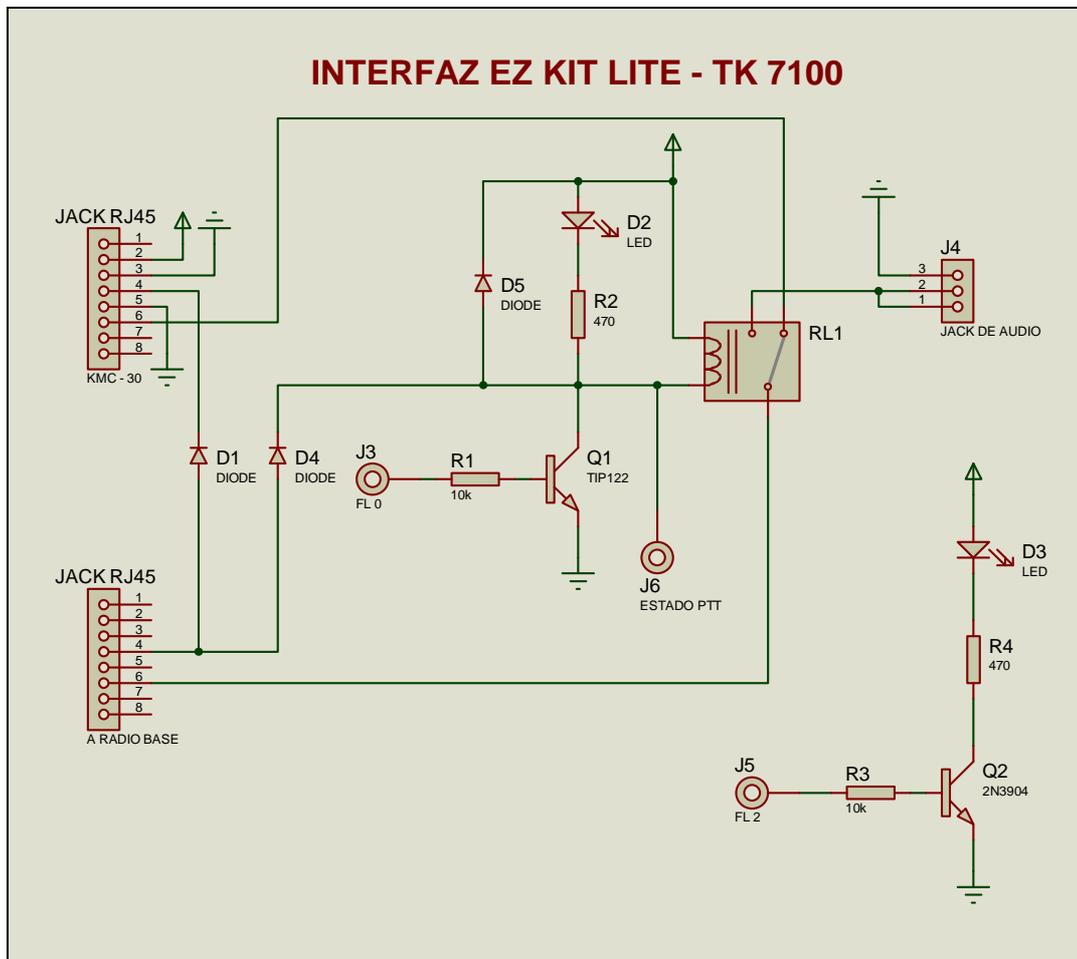


Figura 3. 29: Interfaz tarjeta de desarrollo – radio base

### 3.3.1.2. Operación del circuito

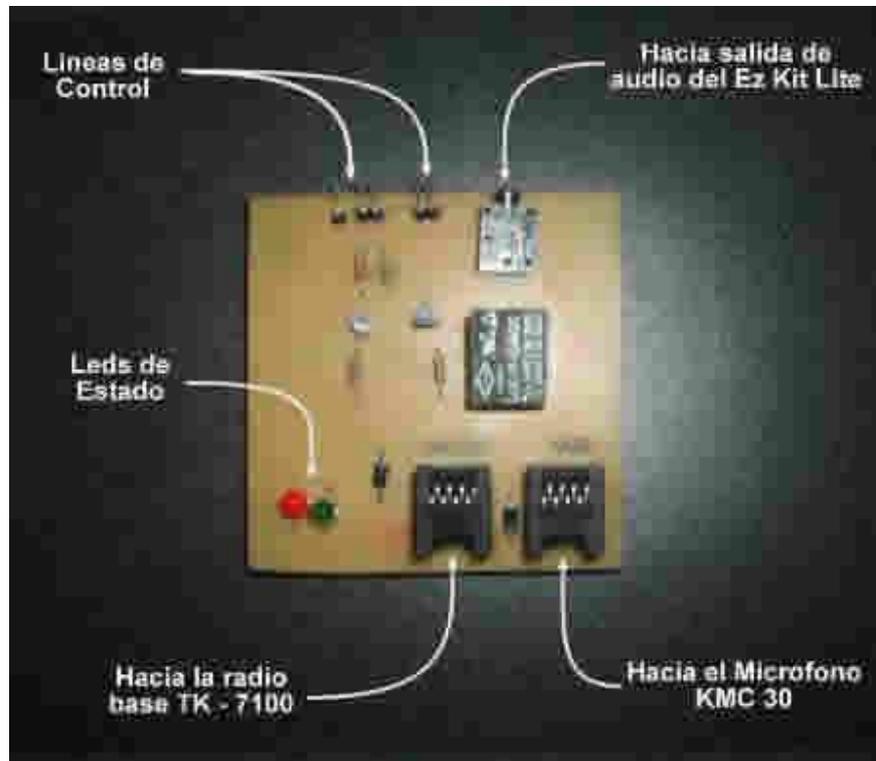
Las entradas FL0 y FL2 corresponden a banderas de salida de la tarjeta de desarrollo Ez Kit lite ADSP 2181 correspondiendo a entradas en el circuito interfaz, la segunda de las mencionadas maneja la frecuencia de parpadeo de un diodo emisor de luz verde (*D3*), una mayor indica el estado de pánico que ha sido activado por el usuario mediante el botón de interrupción externa incorporado en la tarjeta de desarrollo, como se mencionó anteriormente durante este estado la información de localización se envía con mayor regularidad; una menor frecuencia de parpadeo indica un estado normal en el cual la información de localización se envía según lo establecido para los casos de movimiento o reposo.

La selección entre voz y datos se hace por medio de la bandera FL0, la cual está controlada por programa, si se envían cinco voltios a través del pin de dicha bandera se activa entonces un relé (*RL1*) que conmuta del estado normalmente cerrado hacia el origen de los datos de localización, que es precisamente la tarjeta de desarrollo, a través del jack de audio (*J4*); el mencionado estado normalmente cerrado constituye el origen de voz (MIC, Pin 6 del KMC-30); cada vez que se activa el relé se enciende un led rojo (*D2*) señalizando el envío de información de localización.

El par de diodos (*D1*) y (*D4*) hacen posible que solamente una fuente haga la operación de *push to talk* (*PTT*), sea esta de manera automática o mediante el usuario, la primera para transmitir datos y la segunda para voz, de tal suerte que cuando se esté efectuando una la otra se bloquee.

Finalmente (*D5*) es un diodo de protección para el relé y (*J6*) es un pin de salida del circuito interfaz hacia la tarjeta de desarrollo, que indica el estado del *PTT* del usuario, de tal modo que se lo pueda evaluar por programación para evitar la interrupción de transmisión de datos cuando hay transmisión de voz.

La interfaz se la implemento en baquelita y finalmente luce así:



**Figura 3. 30:** Vista Superior Circuito Interfaz

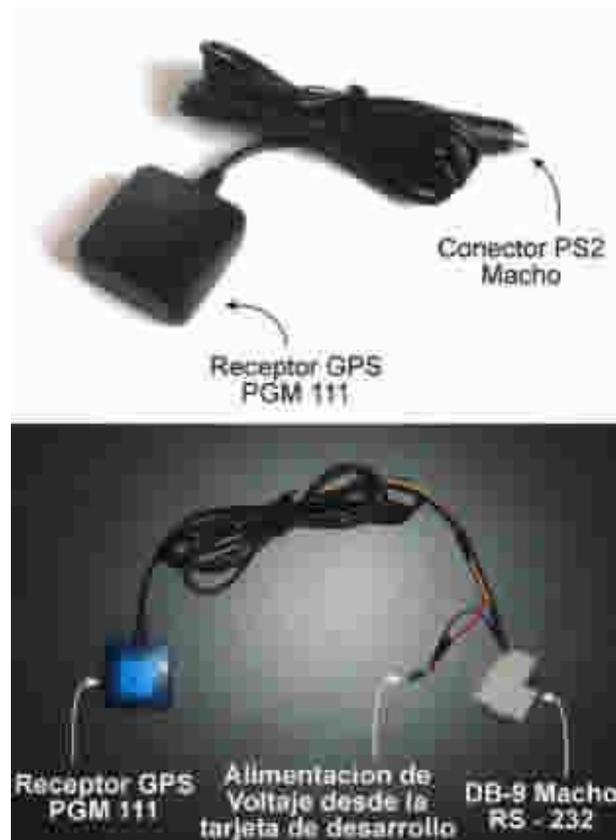


**Figura 3. 31:** Vista lateral Circuito Interfaz

### 3.3.2. INTERFAZ DISPOSITIVO GPS – TARJETA DE DESARROLLO

El dispositivo GPS modelo PGM – 111 originalmente posee una interfaz PS2 para conexión al computador, la misma que debe adaptarse para que el receptor GPS pueda ser conectado a la tarjeta de desarrollo para transmitir datos y obtener alimentación de la misma, de esta manera se reemplazó totalmente la interfaz inicial para poder realizar la conexión de una serial a través de un DB9 macho y headers para obtener la alimentación necesaria; esto se puede realizar por cuanto el receptor provee líneas apropiadas para tal efecto, es decir RS – 232 TX, GND y VCC, que serán las utilizadas.

La siguiente gráfica describe visualmente el resultado final en comparación con lo original:



**Figura 3. 32:** Dispositivo GPS con interfaz serial

### **3.3.3. CONTROL AUTOMÁTICO DE GANANCIA**

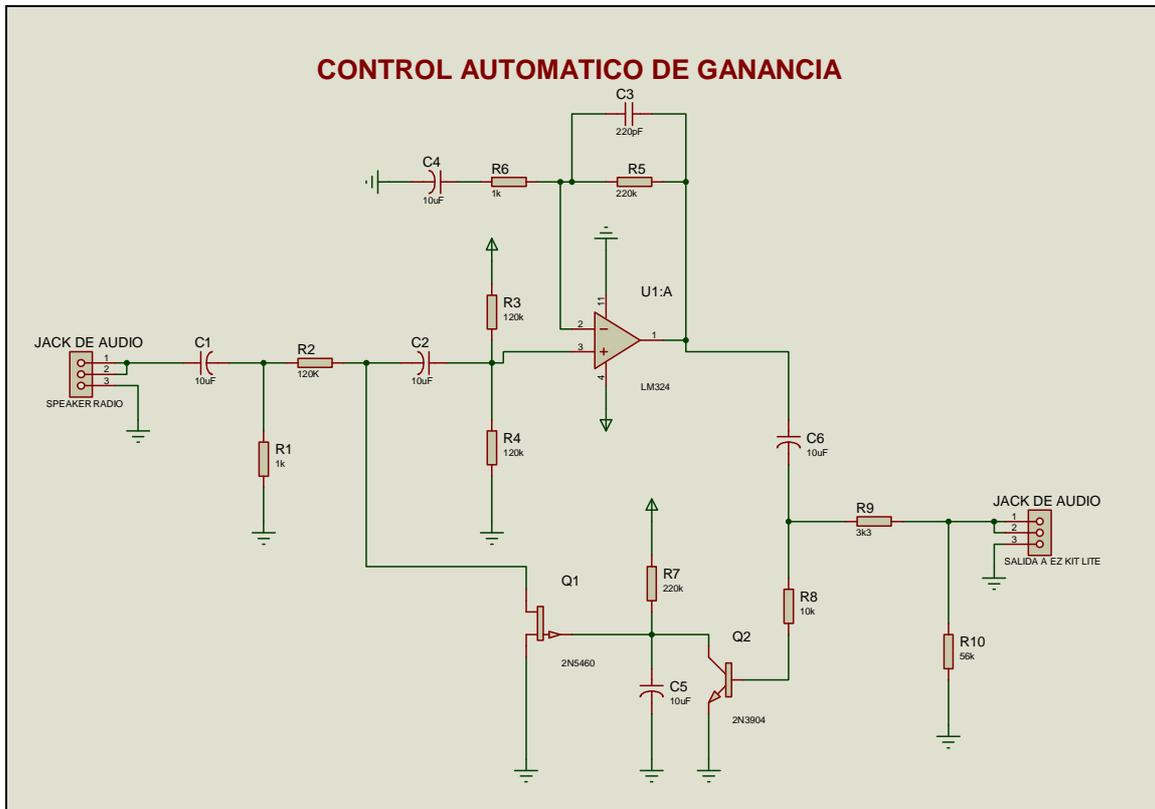
En la sección 2.2.1<sup>72</sup> del capítulo segundo se explicó la necesidad de la construcción de un control automático de ganancia que mantenga el nivel de señal de entrada, al códec en recepción, dentro del rango de valores aceptables por la mismo.

La radio base kenwood TK – 760H usada en recepción posee 32 niveles de volumen, de los cuales sólo un porcentaje bajo sirve para el procesamiento de la señal por la tarjeta de desarrollo; el propósito entonces del control automático de ganancia es hacer que el nivel de señal recibida a la entrada de audio del Kit de desarrollo sea lo más constante posible, siendo así independiente de la mayoría de niveles de volumen que brinda la radio base. Por tanto en la implementación del circuito se debe calibrar para que en la salida del mismo exista un rango pico-pico de voltaje de 0.88 a 1.6 Voltios, el cual fue obtenido experimentalmente, en un proceso que se describe con detalle en el capítulo próximo.

#### **3.3.3.1. Esquemático del circuito**

---

<sup>72</sup> Página 93 del presente proyecto de titulación



**Figura 3. 33:** Control Automático de ganancia

### 3.3.3.2. Operación del circuito

El circuito tiene cuatro etapas principales:

- Entrada de señal
- De ganancia fija
- Etapa de control de entrada de señal
- Salida de señal

La entrada de señal está constituida por un jack de audio estéreo que se encuentra conectado al speaker auxiliar de la radio base, de esta manera se toma la señal proveniente de la radio y se la lleva hacia el circuito.

La etapa de ganancia fija viene dada por un sub-circuito que usa un amplificador operacional ( $U1$ ) y un par de resistencias que la determinan, la cual se calcula como:

$$G_v = 1 + \frac{R5}{R6} = 1 + \frac{220k}{1k} = 221$$

Este valor se mantiene constante a lo largo de la operación del circuito, la etapa que hace que la salida sea la deseada es la denominada de *control de entrada de la señal*, llamada así porque administra el nivel de señal que se deja pasar hacia la etapa de ganancia fija y lo hace a través de un transistor ( $Q1$ ) de efecto campo (JFET) que funciona como una resistencia variable controlada por voltaje. Para que el *JFET* funcione tal como se mencionó, este debe estar polarizado adecuadamente, localizando el punto de operación en la región óhmica y no en la de saturación, para ello está presente el sub-circuito que incluye un transistor de juntura ( $Q2$ ).

Funcionando así, las terminales  $D - S$  de  $Q1$  pueden imaginarse como los extremos variables de un potenciómetro y el voltaje continuo en las terminales  $G - S$  como su perilla, de esta manera se ha implementado un divisor de voltaje, cuya salida es variable a la entrada de la etapa de ganancia fija.

La etapa final es un divisor de voltaje ( $R9$  y  $R10$ ) de donde se toma la señal de salida para llevarla a la tarjeta de desarrollo receptora a través de un jack de audio.

El Control Automático de ganancia se lo implementó en baquelita y finalmente luce así:

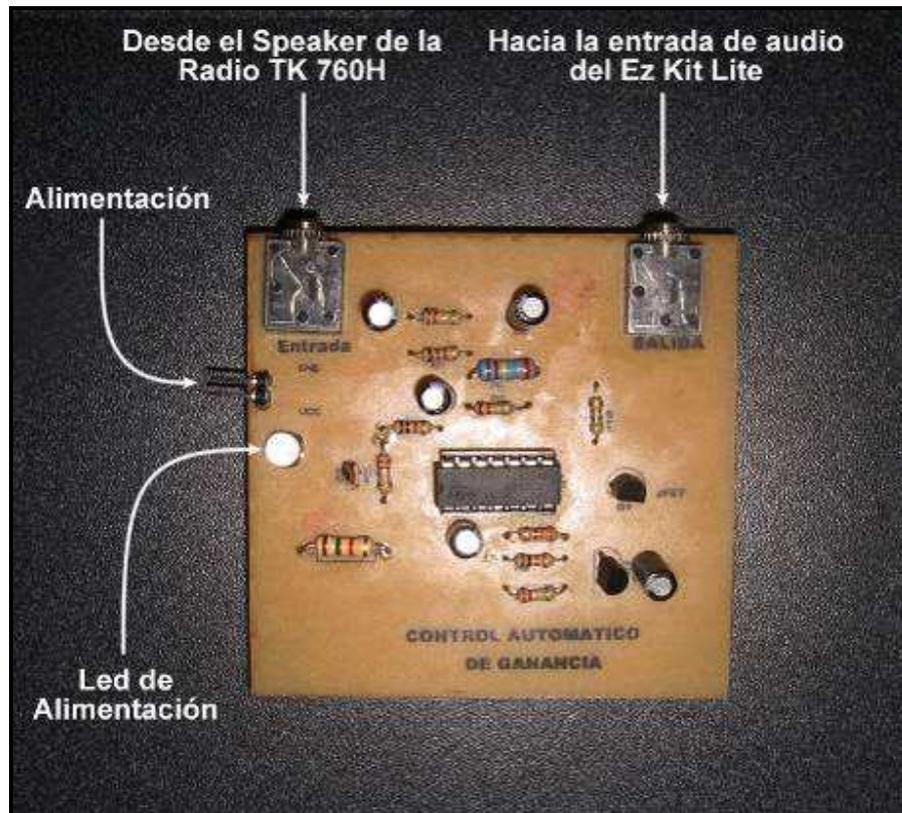


Figura 3. 34: Vista superior circuito control automático de ganancia



Figura 3. 35: Vista Lateral circuito control automático de ganancia

### **3.4. CONSTRUCCIÓN FINAL DE LOS PROTOTIPOS**

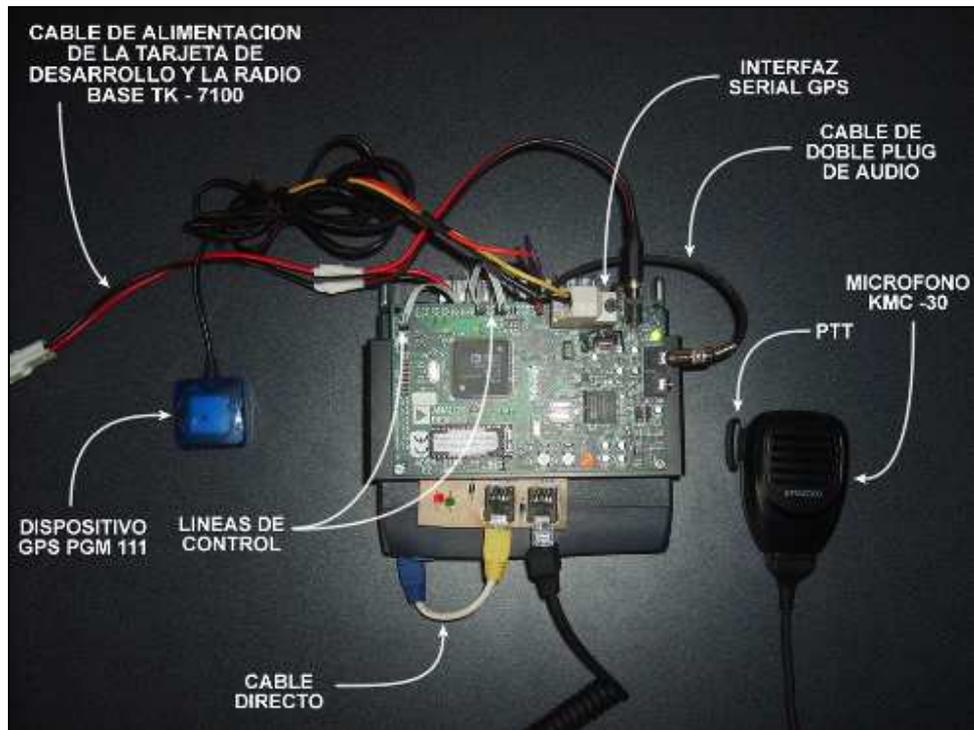
La construcción final de los prototipos es la reunión exitosa de los tres principales componentes que se han analizado a lo largo del presente capítulo, firmware, hardware y software.

#### **3.4.1. TRANSMISOR**

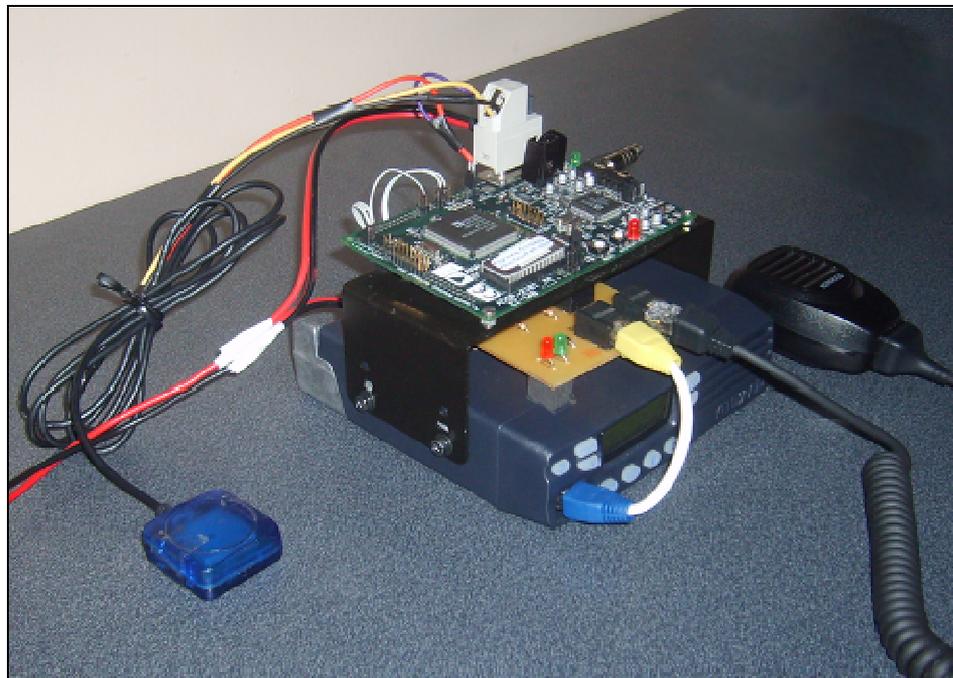
El transmisor integra los siguientes elementos:

- Tarjeta de desarrollo Ez KIT Lite ADSP-2181
- Radio Kenwood TK – 7100
- Interfaz Serial y Dispositivo GPS PGM - 111
- Interfaz tarjeta de desarrollo – radio base
- Antena VHF
- Micrófono y *Push to talk* KMC – 30
- Cable directo

Lo anterior puede ser constatado en las siguientes gráficas ilustrativas:



**Figura 3. 36:** Vista superior prototipo transmisor



**Figura 3. 37:** Vista lateral prototipo transmisor



**Figura 3. 38:** Ambiente de trabajo prototipo transmisor

La tarjeta de desarrollo se montó sobre la radio base utilizando un soporte diseñado para la misma, dejando un espacio adecuado para la inclusión de la interfaz entre ambos elementos, un cable directo conecta la interfaz con la radio base para llevar la información de audio (sea voz o datos) y la señal de PTT hacia esta, el componente KMC – 30 se conecta en la interfaz para brindar la posibilidad de transmitir información de voz, un cable con plugs de audio estéreo en sus extremos se encarga de llevar la información modulada hacia la interfaz desde el kit de desarrollo,

finalmente simples cables conectan las señales de control entre la tarjeta y la interfaz.

### 3.4.2. RECEPTOR

El receptor integra los siguientes elementos:

- Tarjeta de desarrollo Ez KIT Lite ADSP-2181
- Radio Kenwood TK – 760H
- Cable Serial
- Control Automático de Ganancia
- Antena VHF
- Micrófono y *Push to talk* KMC – 30
- Conector “Y” de audio
- Dos cables plug de audio

Lo anterior puede ser constatado en las siguientes gráficas ilustrativas:

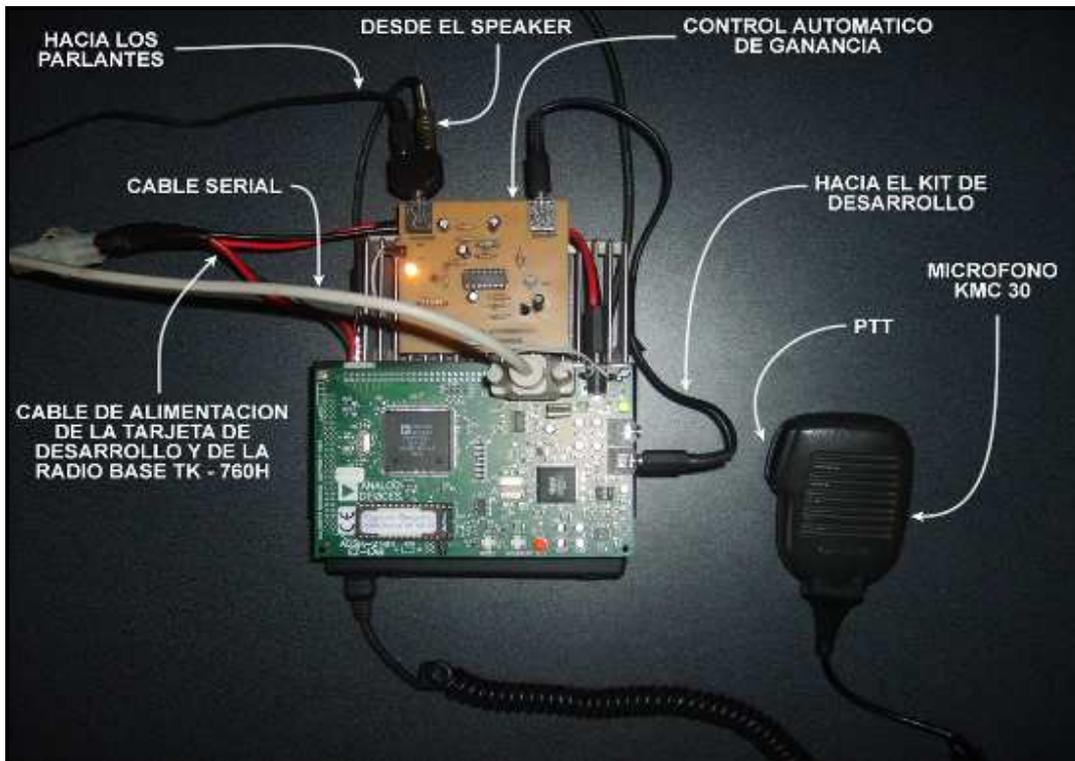


Figura 3. 39: Vista Superior prototipo receptor



Figura 3. 40: Vista lateral prototipo receptor



**Figura 3. 41:** Ambiente de trabajo prototipo receptor

Así como en el transmisor, la tarjeta de desarrollo se montó sobre la radio base a través de un soporte diseñado para la misma, el circuito de control automático de ganancia se ubica en la parte posterior de la radio ya que se dispone de espacio suficiente; el micrófono y PTT KMC – 30 se ubican en la posición original permitiendo una comunicación normal de voz; en la entrada al circuito de control automático de ganancia (CAG) se ubica una “Y” de audio estéreo de modo que pueda aprovecharse la señal proveniente del speaker auxiliar de la radio dos veces, la primera para ser procesada por el circuito CAG y la segunda para llevarse a unos parlantes convencionales de tal manera que la recepción de audio no se altere; la salida del circuito CAG se lleva al kit de desarrollo a través de un cable estéreo; finalmente un cable serial proporciona la información demodulada a un computador.

### **3.5. ESTABLECIMIENTO DE LA SECCIÓN REPETIDORA**

La puesta en marcha de un sistema repetidor es importante por cuanto permite tener una zona de cobertura amplia, lo cual es indispensable en un sistema de localización automática vehicular; se hizo entonces la instalación de un dispositivo repetidor y una antena VHF en la zona oriental de la ciudad de Quito en el sector de Miravalle, de modo que se pueda cubrir buena parte de la capital.

La operación del repetidor en modo semi duplex es bastante sencilla, se necesita configurar el par de frecuencias a usar para transmisión y recepción, tomando en cuenta que la usada por las radio bases para transmitir es en el repetidor considerada como de recepción, y así mismo, la frecuencia de recepción en las radio bases es la de transmisión en el repetidor.

Dichas frecuencias son licenciadas por el ente regulador respectivo de modo que se eviten interferencias con otros sistemas de radio en el servicio fijo móvil terrestre; en las radio bases para transmisión y recepción se usan los valores de 171.275000 y 172.30000 MHz respectivamente.

Las frecuencias señaladas deben programarse tanto en las radio bases como en el repetidor con las observaciones realizadas mediante el software respectivo.

Se instaló un duplexor el cual, como sabemos, permite la operación semi dúplex con una sola antena, es decir aísla el transmisor del receptor utilizando un mismo canal.

El repetidor VHF instalado es el *TKR – 750* de Kenwood.



**Figura 3. 42:** Repetidor TKR – 750 de Kenwood



**Figura 3. 43:** Ambiente funcionamiento del Repetidor



**Figura 3. 44:** Soporte y antena VHF

## **CAPÍTULO 4**

### **REALIZACIÓN DE PRUEBAS Y ANÁLISIS DE RESULTADOS**

El presente capítulo hace una recopilación de las pruebas realizadas a la versión final del sistema básico de localización vehicular implementado, así como también, a las realizadas en su fase de desarrollo. De ser necesario también se describen pruebas efectuadas en algunos de los módulos que componen dicho sistema, estableciendo de esta manera conclusiones que lleven al mejoramiento del desenvolvimiento del sistema total, limiten el alcance del mismo o presenten lineamientos para trabajos futuros.

#### **4.1. VERSIONES DE EVALUACIÓN DE LOS PROTOTIPOS**

La implementación de los primeros prototipos tiene el objetivo de proveer la re - alimentación necesaria para el mejoramiento en versiones futuras.

##### **4.1.1. PRIMERA VERSIÓN DE LOS PROTOTIPOS**

La gráfica mostrada a continuación describe visualmente las características de estos prototipos:

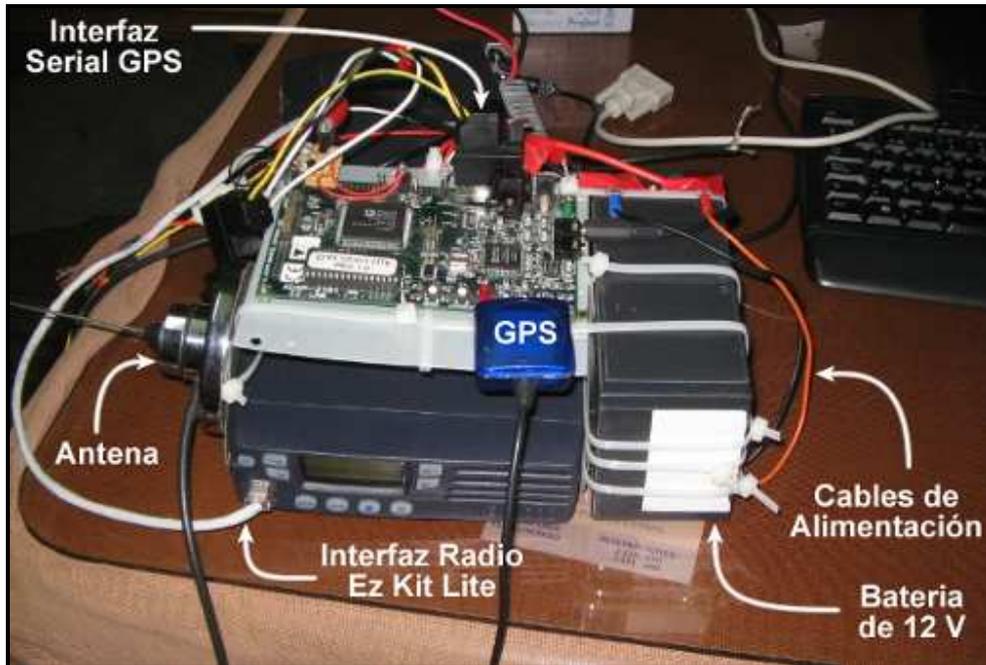


Figura 4. 1: Versión primera prototipo transmisor

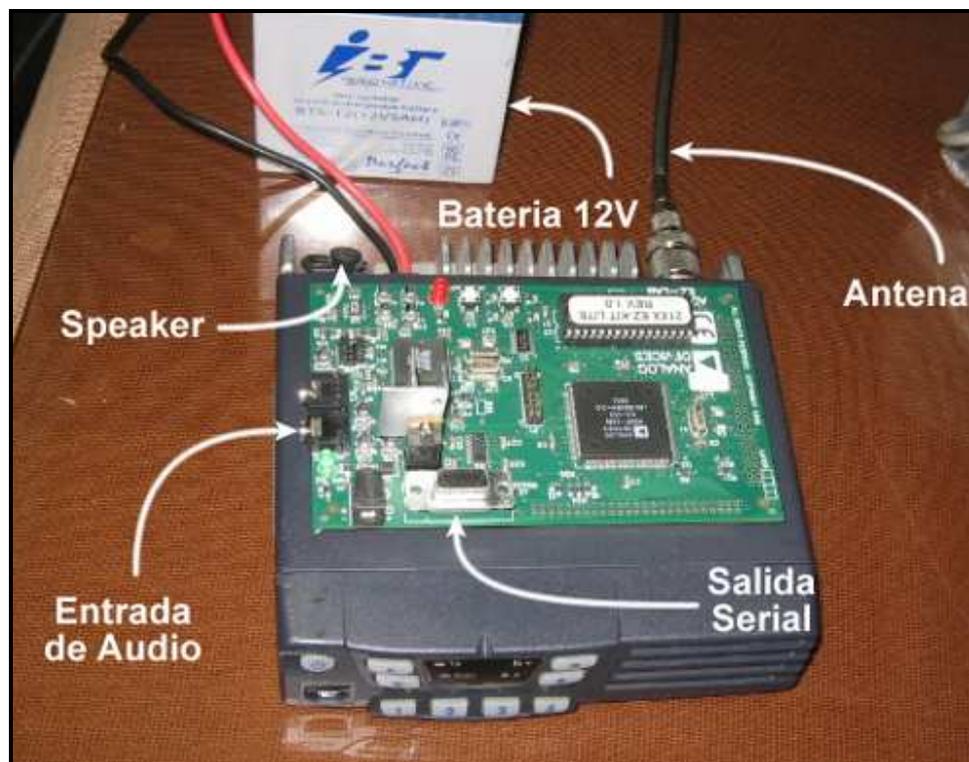


Figura 4. 2: Versión primera prototipo receptor



Figura 4. 3: Primera interfaz tarjeta de desarrollo - radio base

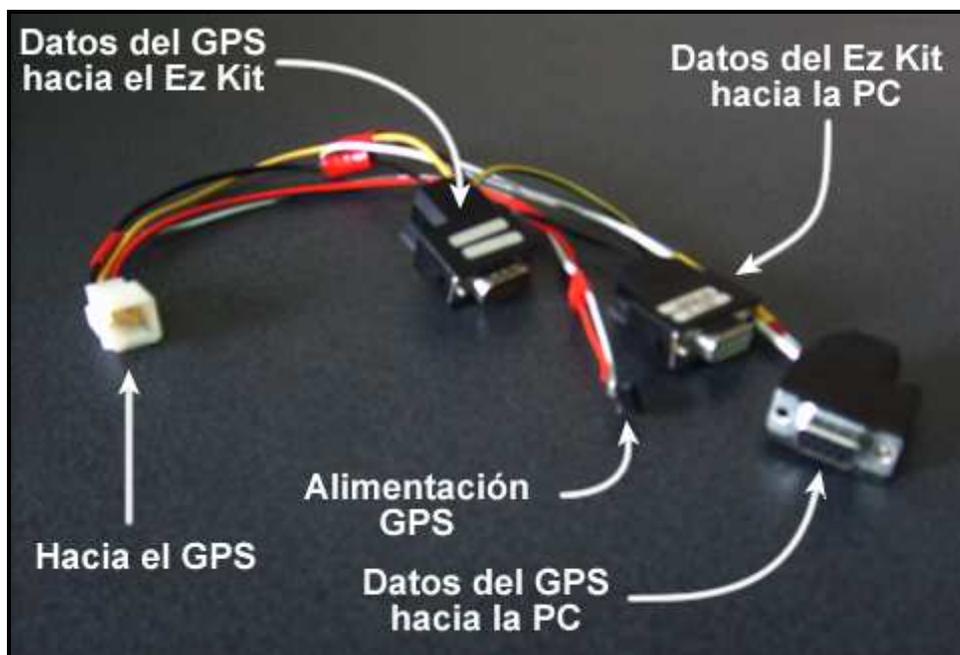


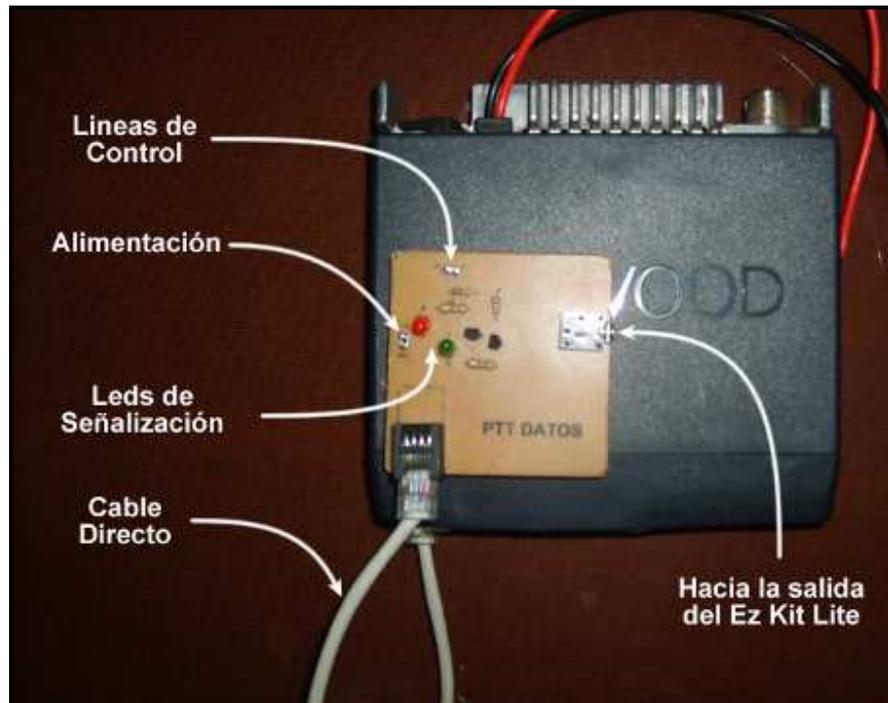
Figura 4. 4: Primera interfaz tarjeta de desarrollo – Receptor GPS

Características generales de ambos prototipos:

- La tarjeta de desarrollo se une a la radio base transmisora a través de una carcasa improvisada.
- En ambos casos (transmisor y receptor) se utilizó una batería de doce voltios para alimentar tanto a la radio base como al Kit de desarrollo, en el caso del transmisor esta se la fijó a la carcasa para poderla transportarla fácilmente, esto es clave por cuanto se necesita tener encendida la tarjeta de desarrollo para no perder al programa en prueba y tener que volver a cargarlo serialmente cada vez.
- La interfaz entre radio base y kit de desarrollo constituye un simple cable fabricado para tener tres terminales, un rj – 45, un plug de audio estéreo y conectores headers para las líneas de control, destinadas para la conexión a la radio base, a la salida de audio de la tarjeta y las banderas de salida respectivamente; se la puede visualizar en la figura 4.3.
- La interfaz tarjeta de desarrollo – receptor GPS, mostrada en la figura 4.4, consta de tres conectores DB9, uno macho para llevar los datos del GPS hacia el kit de desarrollo, uno hembra que se conecta a la PC para comprobar la llegada de dichos datos, otro hembra para permitir la transmisión de datos a la PC desde el EZ Kit Lite. Esta interfaz fue realizada de esta manera para permitir el desarrollo y depuración en la recepción de tramas NMEA.
- El programa cargado ya recepta las tramas GPRMC y las envía periódicamente al receptor sin realizarle ninguna modificación, respecto de una base de tiempo de segundos, mediante un módulo de apertura de canal en fase de desarrollo que aun no toma en cuenta la funcionalidad de voz; el receptor integra un discriminador de bits no inteligente, el cual será explicado en una sección posterior; la velocidad de transmisión es de 600 bps.

#### 4.1.2. SEGUNDA VERSIÓN DE LOS PROTOTIPOS

La gráfica mostrada a continuación describe visualmente las características de este segundo prototipo:



**Figura 4. 5:** Versión segunda prototipo transmisor

El segundo prototipo de receptor fue el finalmente implementado cuyas características específicas pueden ser visualizadas y analizadas con detalle en los numerales 3.1.2<sup>73</sup> y 3.4.2<sup>74</sup> del capítulo tercero.

Características generales de ambos prototipos:

- En relación a los correspondientes primeros prototipos estos ya no necesitan de una batería como las mostradas en las figuras 4.1 y 4.2, existen entonces dos posibilidades de alimentación para el conjunto radio base – Kit de desarrollo, la primera es conectar dicho sistema directamente a la batería del automóvil a

<sup>73</sup> Página 133 del presente proyecto de titulación

<sup>74</sup> Página 174 del presente proyecto de titulación

través de un cable apropiado, con lo cual se pueden realizar las pruebas de campo de una manera más aproximada a la realidad; la segunda es conectar a una fuente de poder simple de laboratorio, esto último con fines de prueba trabajando en modo simplex donde no se necesita de traslado del equipo.

- En esta segunda versión se mejora la interfaz tarjeta de desarrollo – radio base, implementándola completamente en baquelita y añadiéndole diodos emisores de luz para utilizarlos como indicadores de estado, tal como se observa en la figura 4.5.
- En la segunda versión del prototipo receptor se añade un circuito de control automático de ganancia que permite satisfacer la consideración de diseño especificada en el numeral 2.3 del capítulo segundo.
- El programa cargado trae consigo un módulo de apertura automática de canal mejorado, basando su funcionamiento en parámetros de velocidad y tiempo pero que aun no toma en cuenta la funcionalidad de voz, además el discriminador en receptor se lo diseña para una operación inteligente, se modifica la trama a enviar para incluir caracteres de inicio y fin de la misma, mediante los ASCIIs > y < respectivamente; la velocidad de transmisión sigue siendo de 600 bps.

#### **4.1.3. TERCERA VERSIÓN DEL PROTOTIPO TRANSMISOR**

La tercera versión del prototipo es la finalmente implementada cuyas características específicas pueden ser analizadas y visualizadas con detalle en los numerales 3.1.1<sup>75</sup> y 3.4.1<sup>76</sup> respectivamente

Características generales:

- El prototipo final del transmisor trae consigo una evolucionada interfaz tarjeta de desarrollo – radio base permitiendo la coexistencia física entre las transmisiones

---

<sup>75</sup> Página 106 del presente proyecto de titulación

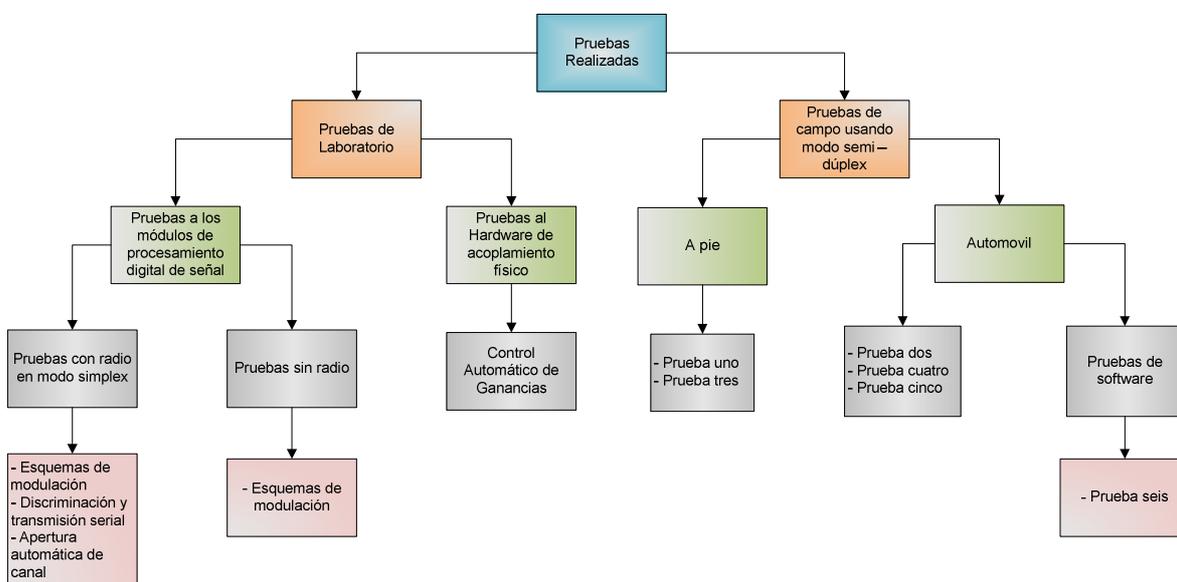
<sup>76</sup> Página 171 del presente proyecto de titulación

de voz e información de navegación, tal como se lo describe en el numeral 3.1.1.4<sup>77</sup>, ya que las anteriores versiones no lo permitían hacer.

- El módulo de evaluación de apertura automática del canal se mejora aun más permitiendo la coexistencia de voz e información de navegación, sin interferencia entre ambas y con prioridad de la primera, además de no permitir la pérdida del “turno” de transmisión de información por prioridad de voz, controlando de esta manera a la interfaz mencionada en el párrafo anterior, se adiciona además a este módulo la transmisión por estado de pánico; se implementa un control de validación de datos por firmware generando un checksum en transmisión y comprobándolo en recepción, la velocidad de transmisión se modifica a la característica de diseño de 1200 bps.

## 4.2. TIPOS DE PRUEBAS REALIZADAS

El desarrollo de un sistema de comunicaciones aplicado exige la realización de varias pruebas en todos los puntos esenciales que lo componen, en los numerales siguientes se trata de detallarlas de la manera más clara posible.



**Figura 4. 6:** Diagrama de bloques de pruebas realizadas

<sup>77</sup> Página 117 del presente proyecto de titulación

#### **4.2.1. PRUEBAS DE LABORATORIO**

Constituyen el conjunto de pruebas realizadas en condiciones controladas para evaluar el funcionamiento de los elementos de firmware y hardware de acoplamiento físico, con lo cual se obtiene la retroalimentación respectiva para el desarrollo y depuración de las etapas del sistema mencionadas.

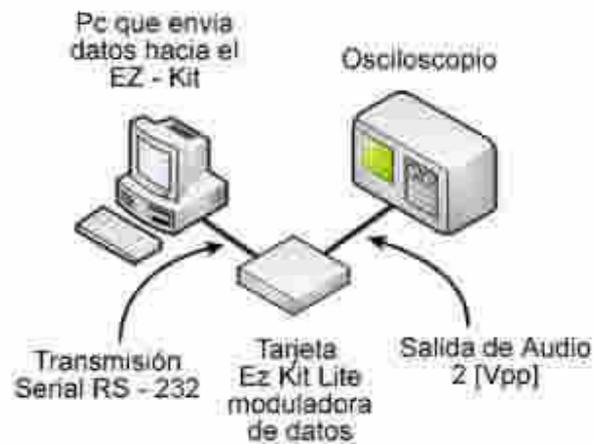
##### **4.2.1.1. Pruebas a los módulos de procesamiento digital señal**

El desarrollo y comprobación de funcionamiento de los diferentes módulos que componen el sistema fue realizado en principio sin la inclusión de las radio bases, es decir solamente con la tarjeta de desarrollo, posteriormente las pruebas se desarrollaron en el ambiente de radiofrecuencia mediante el uso de radio bases, pero en configuración simplex, o sea utilizando una misma frecuencia para transmisión y recepción, la razón para no hacer uso del modo semi dúplex es que para ello se debe conectar las respectivas antenas para alcanzar el repetidor, causando interferencia entre las mismas debido a la cercanía mutua, lo cual no sucede en modo simplex donde no es necesario la conexión de elementos radiantes.

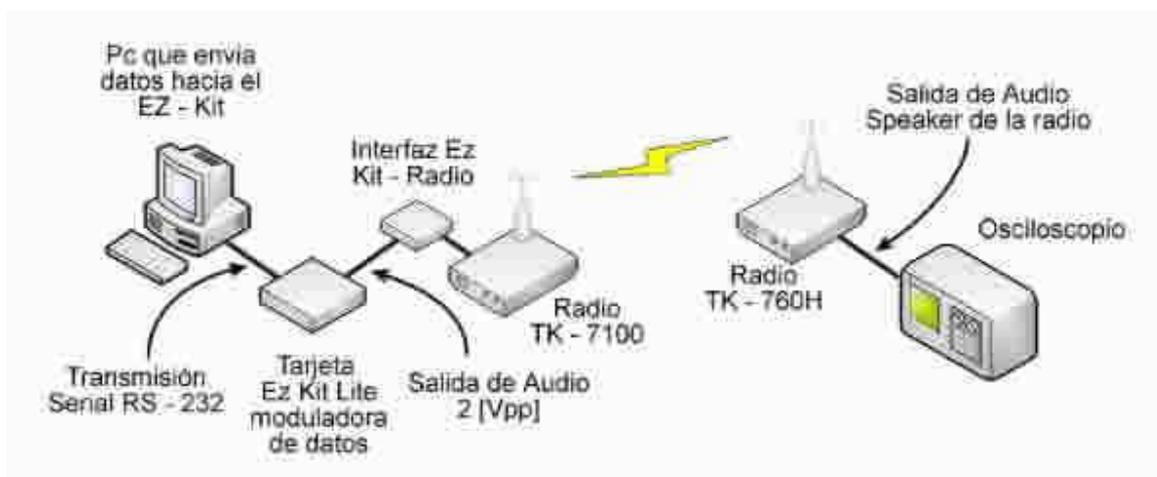
##### ***4.2.1.1.1. Pruebas a los esquemas de modulación***

Las pruebas a los diferentes esquemas de modulación verifican la validez en el diseño de los respectivos algoritmos, tomando en cuenta como modulante una secuencia de bits conocida, las pruebas son realizadas con y sin la inclusión de radio bases.

Los algoritmos implementados fueron ligeramente modificados para que se module una secuencia de bits seleccionada previamente y enviada hacia la tarjeta de desarrollo mediante una interfaz serial a través de un programa terminal.



**Figura 4. 7:** Esquema de pruebas a los tipos de modulación sin radio



**Figura 4. 8:** Esquema de pruebas a los tipos de modulación vía radio en modo simplex

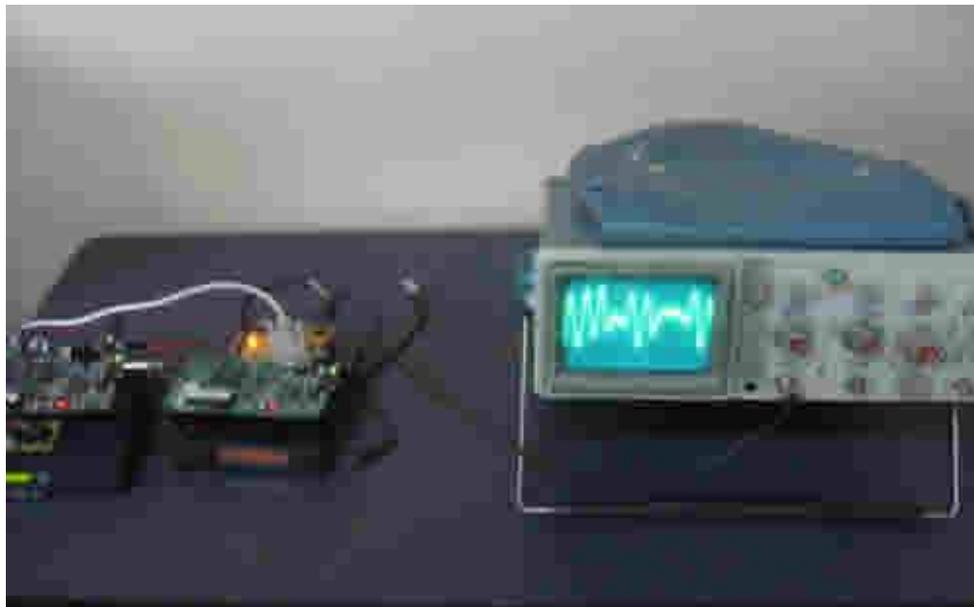
#### 4.2.1.1.1. Modulación y Demodulación ASK

El resultado de la modulación ASK puede verificarse mediante la siguiente ilustración real:



**Figura 4. 9:** Forma de onda ASK

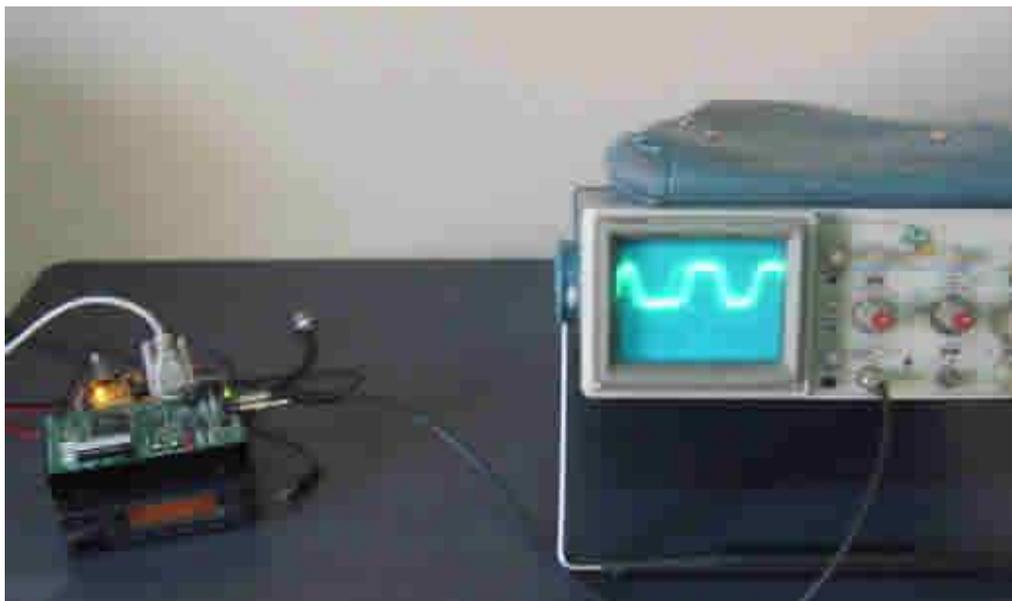
La secuencia de bits modulada y mostrada en la figura anterior corresponde a 0010101, es claro darse cuenta que la señal mostrada no ha atravesado el canal inalámbrico de comunicaciones.



**Figura 4. 10:** Forma de onda ASK que ha atravesado el canal

La figura 4.10 describe una forma de onda ASK que ha atravesado el canal de comunicaciones inalámbrico para lo cual se ha utilizado las radio bases operando en modo simplex, nótese el efecto que produce dicho canal sobre esta señal comparándola con su contraparte en la figura 4.9, es además claro darse cuenta que lo que se necesita para demodular dicha señal es un filtro pasa bajos que actúe como detector de envolvente.

La figura que se presenta a continuación muestra la salida que se obtiene después del filtro pasa bajos citado en el párrafo anterior, donde puede visualizarse además los efectos de la aplicación de la función valor absoluto, una señal cuasi binaria enteramente positiva de donde habrá que discriminar los bits de datos, para un mayor detalle del proceso demodulatorio se puede referir al numeral 3.1.2.2.1<sup>78</sup>



**Figura 4. 11:** Señal cuasi binaria obtenida de ASK

#### **4.2.1.1.1.2. Modulación y Demodulación FSK**

El resultado de la modulación FSK puede verificarse mediante la siguiente ilustración real:

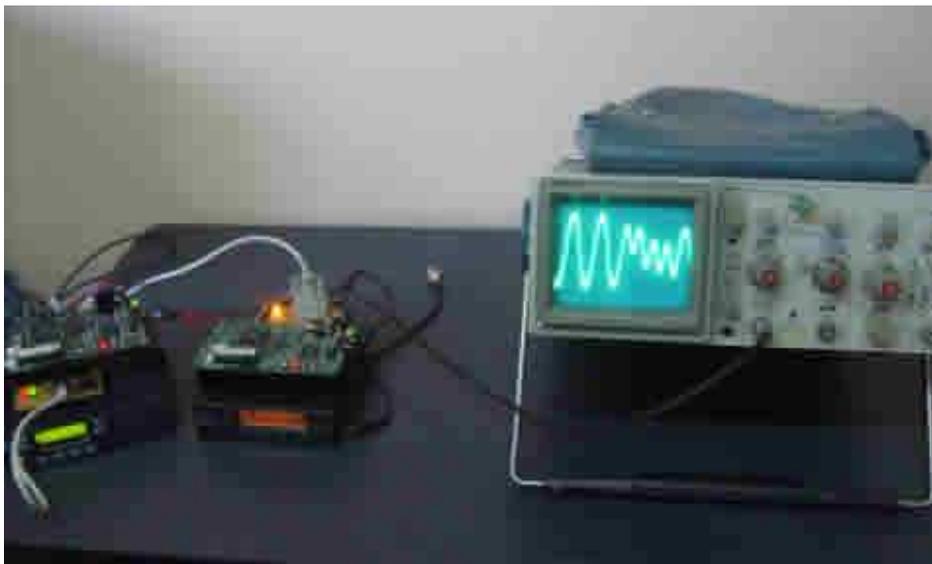
---

<sup>78</sup> Página 137 del presente proyecto de titulación



**Figura 4. 12:** Forma de onda FSK

La secuencia de bits modulada y mostrada en la figura anterior corresponde a 00011, donde la frecuencia más alta corresponde a un cero lógico y la más baja a un uno lógico, el espacio visible luego de los dos primeros ceros se justifica por cuanto se configura el programa terminal para repetir infinitamente la secuencia señalada, de modo que dicho espacio constituye una transición en la repetición. Es claro darse cuenta que la señal mostrada no ha atravesado el canal inalámbrico de comunicaciones ya que no se visualizan los efectos del mismo.



**Figura 4. 13:** Forma de onda FSK que ha atravesado el canal

La figura 4.13 claramente muestra una forma de onda FSK que ha atravesado el canal inalámbrico de comunicaciones por medio de la utilización de las radio bases en configuración simplex, nótese que el canal afecta a la frecuencia más alta con una mayor atenuación, además se puede notar claramente que la demodulación no coherente puede ser efectuada sencillamente aplicando un filtro pasa banda que deje pasar solamente la baja frecuencia, obteniendo de esta manera una señal parecida ASK, con lo cual resta solamente aplicar la función valor absoluto y un filtro pasa bajos.

Efectuando el proceso de demodulación descrito en el párrafo anterior se obtiene la siguiente señal:



**Figura 4. 14:** Señal cuasi binaria obtenida de FSK

De la figura anterior puede deducirse exactamente lo mismo que lo inferido para la correspondiente figura en ASK.

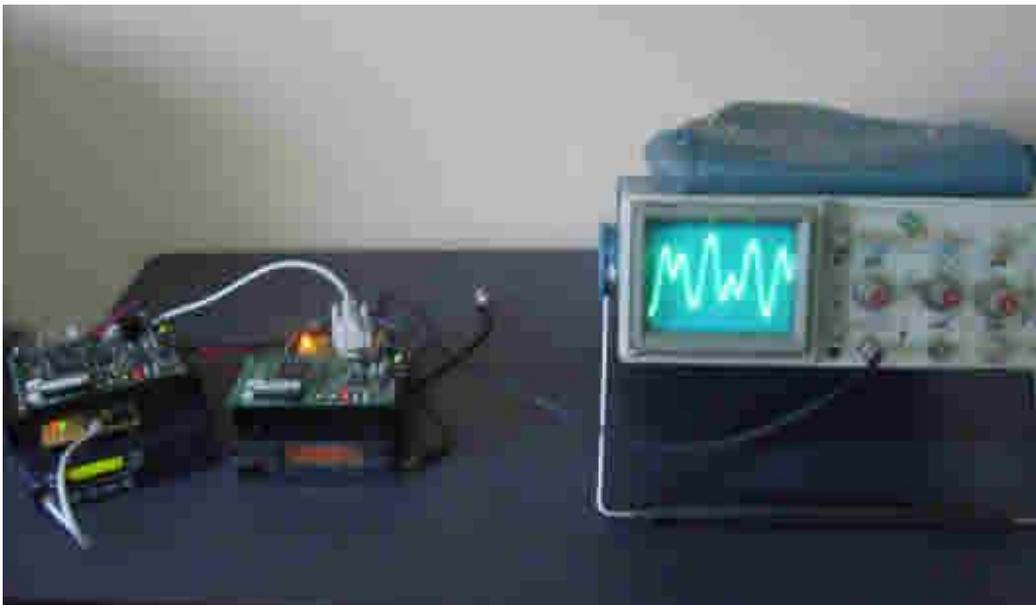
#### **4.2.1.1.1.3. Modulación y Demodulación PSK**

El resultado de la modulación PSK puede verificarse mediante la siguiente ilustración real:



**Figura 4. 15:** Forma de onda PSK

La secuencia de bits modulada y mostrada en la figura anterior es 1010, evidentemente la señal mostrada aún no ha atravesado el canal inalámbrico de comunicaciones.



**Figura 4. 16:** Forma de onda que ha atravesado el canal

En la imagen presentada anteriormente se puede verificar los efectos introducidos por el canal de comunicaciones, la tendencia observada en los tres esquemas de modulación es principalmente los cambios irregulares de amplitud en la señal de salida del speaker auxiliar de la radio receptora, desde donde se la toma para su demodulación, esto explica la baja calidad en la radiodifusión AM en donde se presencia el mismo fenómeno.

Es en este contexto donde la ventaja diferenciadora de la modulación PSK cobra mayor relevancia, ya que la información que lleva está en función de su fase y no de su amplitud, ofreciendo de esta manera un mayor blindaje contra los efectos del ruido en el canal de comunicaciones inalámbrico.

La ventaja mencionada en el párrafo anterior tiene un costo, el proceso de demodulación presenta un mayor grado de complejidad que para los otros esquemas, por tratarse este de un procedimiento coherente. Dicho proceso se lo ha implementado y su resultado es el mostrado a continuación:



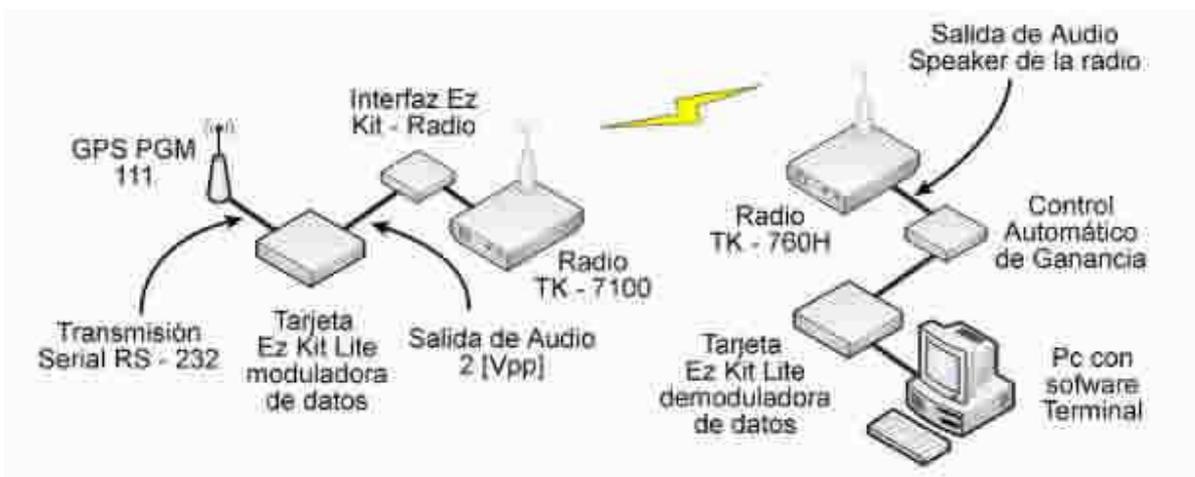
**Figura 4. 17:** Señal cuasi binaria obtenida de PSK

Obsérvese la diferencia de la figura 4.15. con sus correspondientes 4.11 y 4.14, es claro darse cuenta que la demodulación PSK no altera el rango numérico usado en la

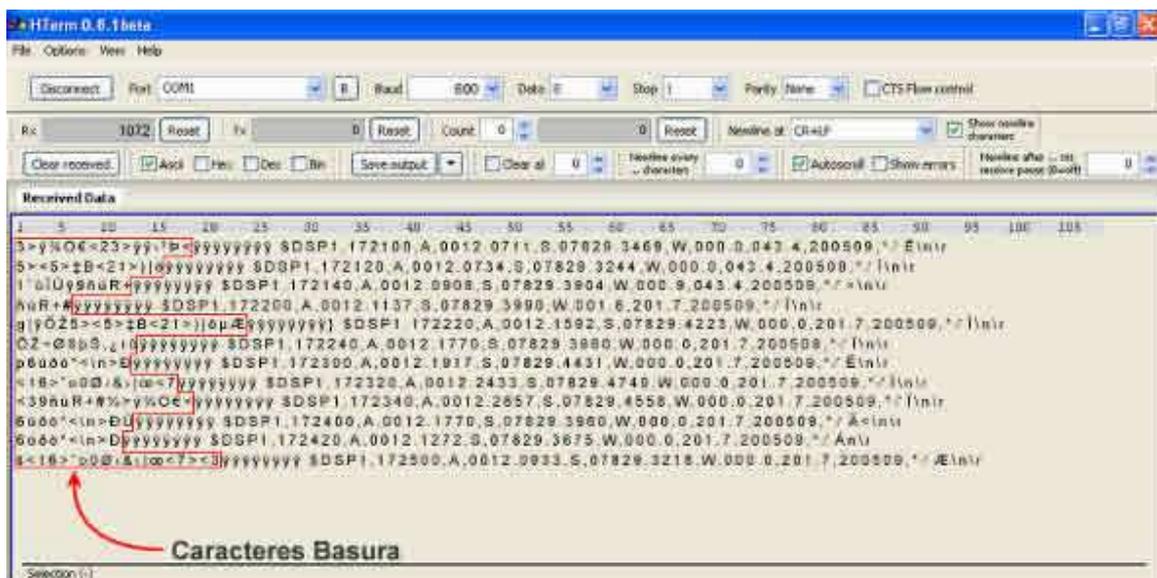
tarjeta de desarrollo, sigue siendo utilizado en su totalidad y no en la mitad como sucede en ASK y FSK.

**4.2.1.1.2. Pruebas a la sección de discriminación y transmisión serial**

El par de módulos mencionados en el título del presente numeral no pueden ser evaluados de manera independiente, se necesita pues de la transmisión serial para conocer si los bits han sido discriminados correctamente.



**Figura 4. 18:** Esquema de prueba a la sección de discriminación y transmisión serial



**Figura 4. 19:** Discriminador de bits no inteligente

En principio se implementó un discriminador de bits no inteligente, es decir este se encargaba de establecer el estado lógico de los cuasi bits en la señal pseudo binaria y enviar inmediatamente hacia la etapa de transmisión serial, el resultado es el mostrado en la figura 4.19, de donde se puede inferir:

- El canal de comunicaciones se abre unos instantes antes de depositar la información, en este pequeño instante se introduce ruido propio del mismo, el receptor solamente demodula la señal correspondiente a la trama enviada, sin embargo discrimina este ruido como si se tratasen de datos válidos lo que es incorrecto, esto explica los caracteres “basura” que se hallan antes del inicio de la trama.
- Se pueden observar además los caracteres de entrenamiento enviados, los cuales se los puede identificar con el ASCII `ÿ` y también el espacio en blanco dejado entre estos nótese además que aun no se implementan los caracteres de inicio y fin de trama `>` y `<` respectivamente.
- No se observa caracteres “basura” al final de la trama ya que el canal de comunicaciones inalámbrico se cierra automáticamente después de enviado el último carácter (`lr`) miembro de la trama enviada.

Se puede deducir entonces que con este mismo discriminador, si se usa la transmisión de voz en los instantes de silencio de información de localización, en el lado receptor la transmisión serial reportará una gran cantidad de caracteres “basura”, lo que una vez más, es inadecuado.

La solución es implementar un discriminador de bits inteligente que se entere de lo que le llegue y así poder o no transmitirlo serialmente hacia el computador, es precisamente lo que se realizó en la implementación final, lo cual puede verse con mayor detalle en el numeral 3.1.2.3<sup>79</sup> del capítulo tercero.

---

<sup>79</sup> Página 146 del presente proyecto de titulación

La figura mostrada a continuación incluye las correcciones efectuadas en el respectivo algoritmo, incluyéndose además una sección de chequeo del checksum enviado por el transmisor, pensado inicialmente para ser verificado por software, de modo que la información se valida ahora por firmware.

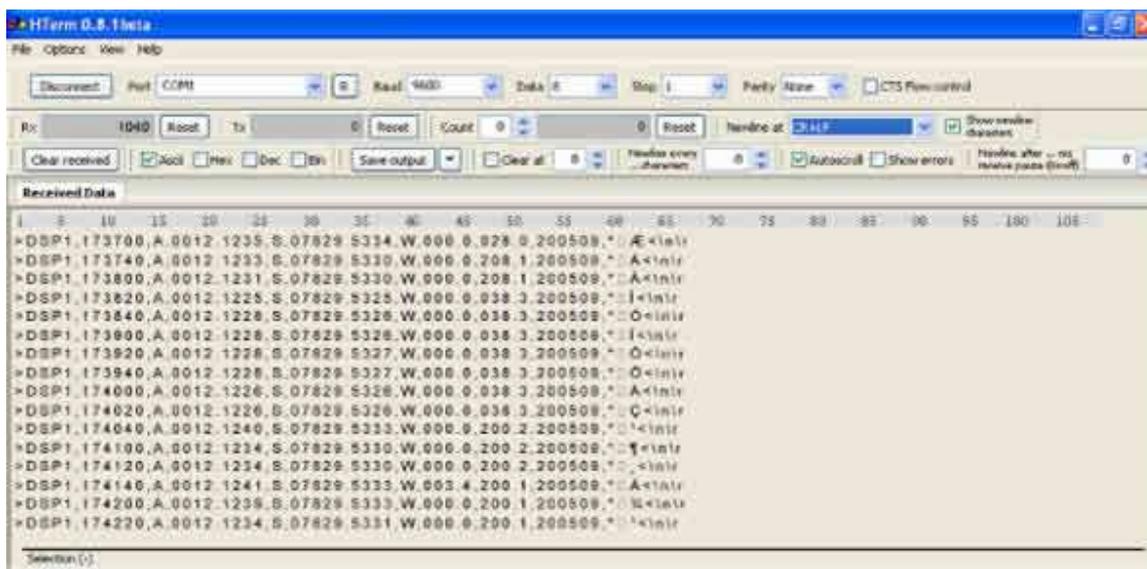


Figura 4. 20: Discriminador de bits inteligente

El software Terminal usado es el *HTerm*, el cual permite realizar las configuraciones usuales de puerto serial como velocidad de transmisión, número de bits de datos, bits de parada, control de flujo y otras más que pueden ser visualizadas en las figuras anteriores y que el típico Hyper terminal no lo permite fácilmente.

#### 4.2.1.1.3. Pruebas al módulo de Apertura Automática del Canal

Una vez más para esta evaluación se hizo uso del software terminal mencionado anteriormente.

La primera prueba consiste en verificar si en estado de pánico, la transmisión y recepción de tramas se realiza cada veinte segundos, tiempo consensuado para dicho estado, recordando además que este es independiente del movimiento del vehículo.

La figura mostrada a continuación describe el desenvolvimiento de dicha prueba:

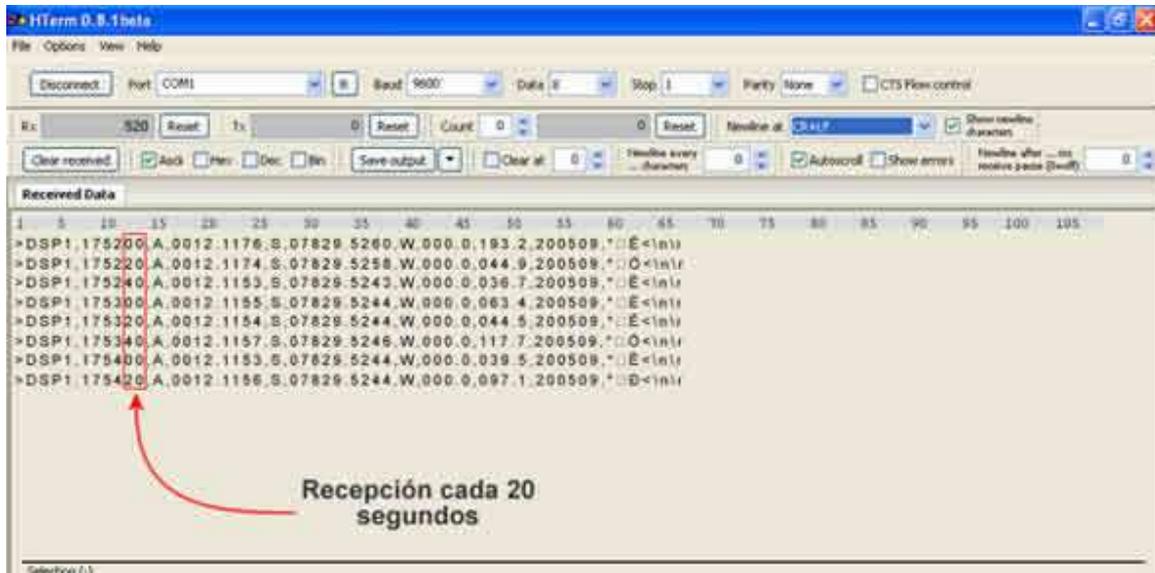


Figura 4. 21: Recepción de tramas en estado de pánico

La segunda prueba verifica si la transmisión y recepción de la trama en estado de movimiento se la realiza cada minuto, tiempo consensuado para dicho estado, la figura mostrada a continuación describe la operación del sistema:

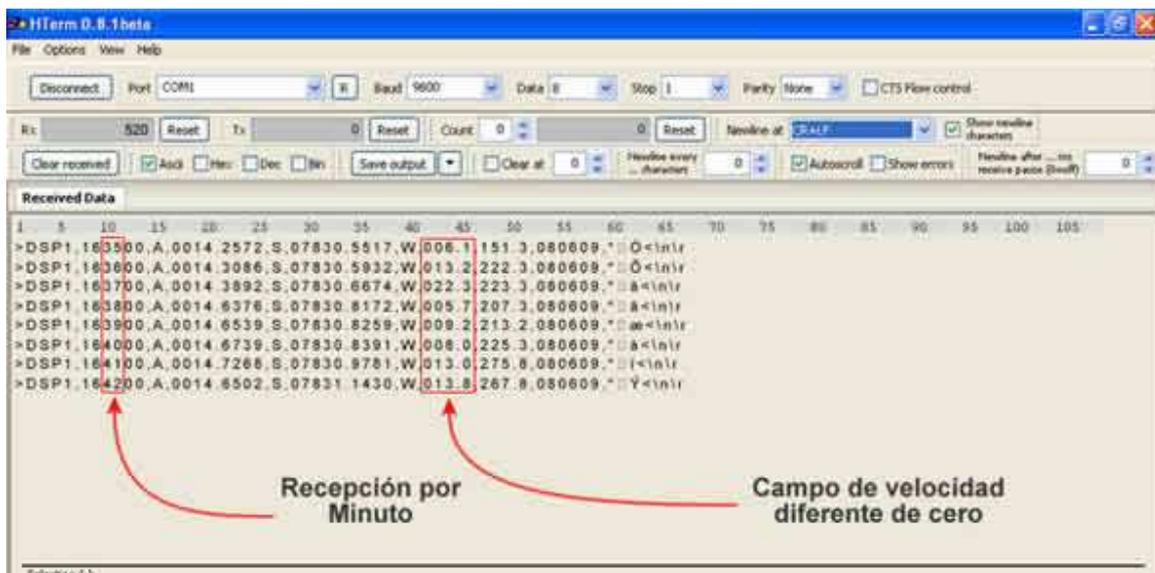


Figura 4. 22: Recepción de tramas en estado de movimiento

La tercera prueba constata si en estado de reposo la transmisión y recepción de tramas se realiza cada dos minutos, tiempo consensuado para dicho estado, la figura mostrada a continuación describe el proceso:

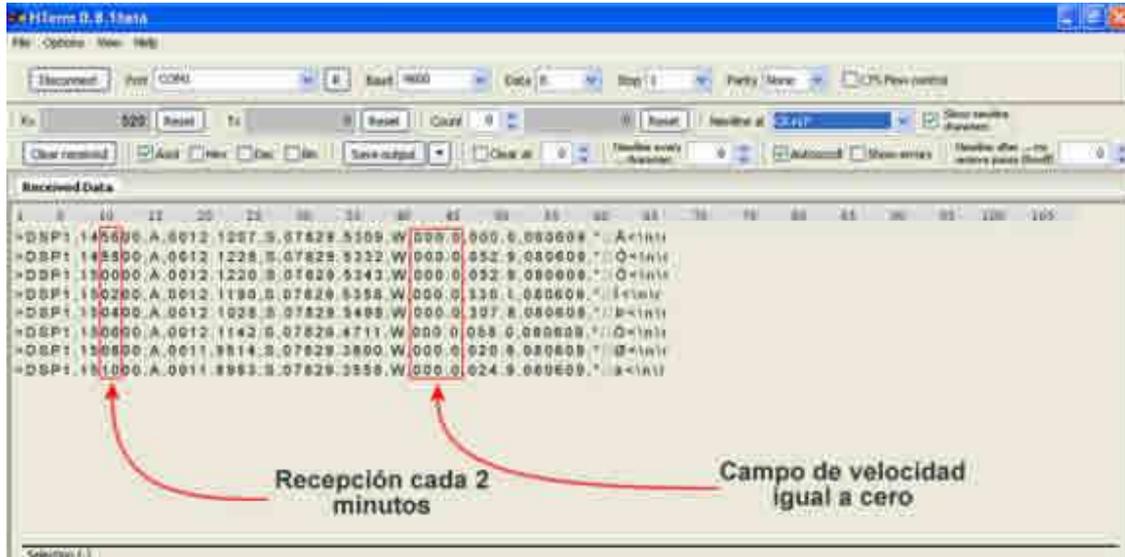


Figura 4. 23: Recepción de tramas en estado de reposo

La siguiente prueba final verifica si, habiéndose cancelado el envío de datos debido a que el usuario ocupa el canal con una transmisión de voz, se envía una trama inmediatamente después de que el usuario termina su comunicación.

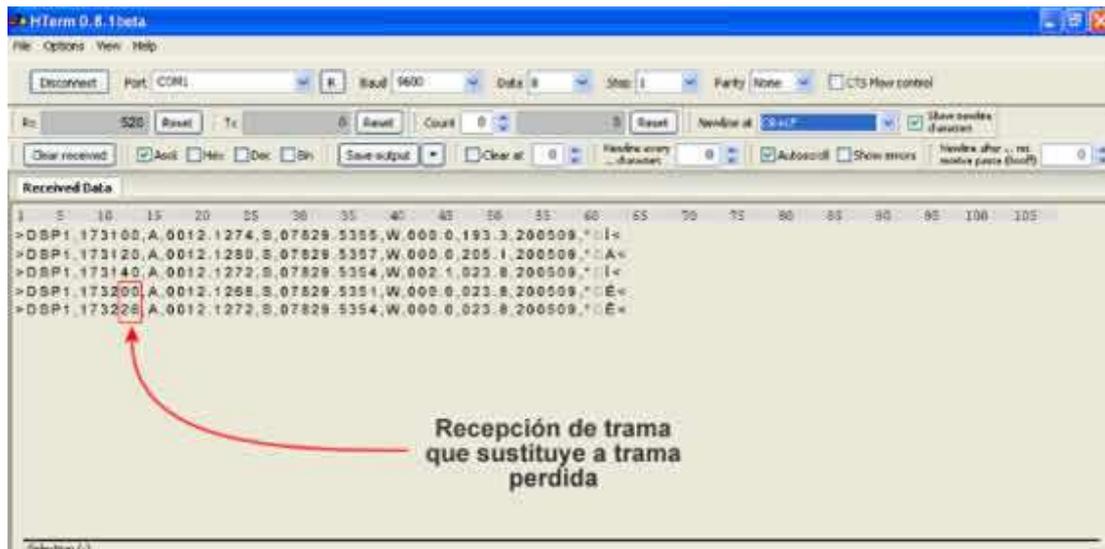


Figura 4. 24: Sustitución de trama perdida

Como puede comprobarse en la gráfica anterior, la trama cuya transmisión fue cancelada corresponde al tiempo 173220, el usuario termina su comunicación en el momento 173225 y la transmisión de la trama sustituta ocurre en 173226.

De esta manera se comprueba lo especificado para el módulo de apertura automática del canal de comunicaciones especificado en el numeral 3.1.1.4<sup>80</sup> del capítulo tercero.

#### **4.2.1.2. Pruebas al hardware de acoplamiento físico**

Las pruebas de Hardware son referidas únicamente a los circuitos implementados físicamente en baquelita, permitiendo saber su comportamiento práctico para determinar algún cambio que haya que efectuarse en pro de los objetivos perseguidos al hacer realidad dicho circuito.

##### ***4.2.1.2.1. Control Automático de ganancia***

En el numeral 2.2.1<sup>81</sup> del capítulo segundo se hizo referencia a una necesidad de implementar un control automático de ganancia, en el apartado 3.3.3<sup>82</sup> del tercer capítulo se explica el funcionamiento del circuito ya implementado y su rol en el sistema receptor, la presente sección completa el estudio describiendo las pruebas de su funcionamiento y explicando en forma más detenida la evaluación de la cual se obtuvo el rango de salida al cual se lo calibró.

Una primera etapa consistió en verificar el número de pasos de volumen disponibles en la radio receptora y el voltaje pico – pico asociado a cada uno de estos, para ello se envió normalmente la trama por la radio transmisora y se la receptó con la radio correspondiente, la salida del speaker auxiliar de esta se la conecta directamente al osciloscopio a través de un conector “Y” estéreo.

---

<sup>80</sup> Página 117 del presente proyecto de titulación

<sup>81</sup> Página 97 del presente proyecto de titulación

<sup>82</sup> Página 167 del presente proyecto de titulación

La segunda etapa, simultanea a la primera, consiste en demodular la señal recibida por radio y comprobar, en cada paso de volumen, si se la recibe correctamente mediante el software terminal, para esto se conecta la entrada de la tarjeta al speaker auxiliar de la radio receptora a través de la mencionada “Y”.

<b>Pasos de Volumen</b>	<b>Voltaje Pico – Pico [V]</b>	<b>Resultado Sin AGC [% tramas correctas]</b>	<b>Resultado Con AGC [% tramas correctas]</b>
1 (Menor Volumen)	0.04	0	0
2	0.09	0	0
3	0.2	0	100
4	0.28	0	100
5	0.44	0	100
6	0.56	0	100
7	0.72	0	100
8	0.88	100	100
9	1.0	100	100
10	1.2	100	100
11	1.4	100	100
12	1.6	100	100
13	1.8	0	100
14	2.0	0	100
15	2.2	0	100
16	2.4	0	100
17	2.8	0	100
18	3.2	0	100
19	3.6	0	100
20	4.0	0	100
21	4.4	0	100
22	4.8	0	100
23	5.2	0	100
24	5.6	0	100
25	6.0	0	100
26	6.4	0	100
27	6.8	0	100
28	7.2	0	100
29	7.6	0	100
30	9.6	0	100
31	12	0	100
32 (Mayor Volumen)	12	0	100

**Tabla 4. 1:** Prueba circuito control automático de ganancia

El resultado de la prueba efectuada se la verifica en la tercera columna de la tabla 4.1, de esta manera se estableció el rango de 0.88 a 1.6 donde se obtuvo el cien por ciento de tramas recibidas correctamente.

La salida del circuito de control automático de ganancia se calibró para que esté dentro de este rango, concretamente en el valor de 1.2 voltios pico – pico, la prueba descrita párrafos arriba se repitió con el circuito implementado y conectado adecuadamente, el registro de la misma corresponde a la columna cuarta de la tabla 4.1, donde evidentemente el lector se dará cuenta de la efectividad del control automático de ganancia.

Cuantitativamente, sin el circuito de control automático de ganancia se tenía aproximadamente el quince por ciento de niveles de volumen aptos para recepción, con dicho circuito se obtiene el noventa y cuatro por ciento de niveles de volumen aptos, con lo cual el objetivo en la construcción de dicho circuito se ha cumplido.

#### 4.2.2. PRUEBAS DE CAMPO EN MODO SEMI - DÚPLEX

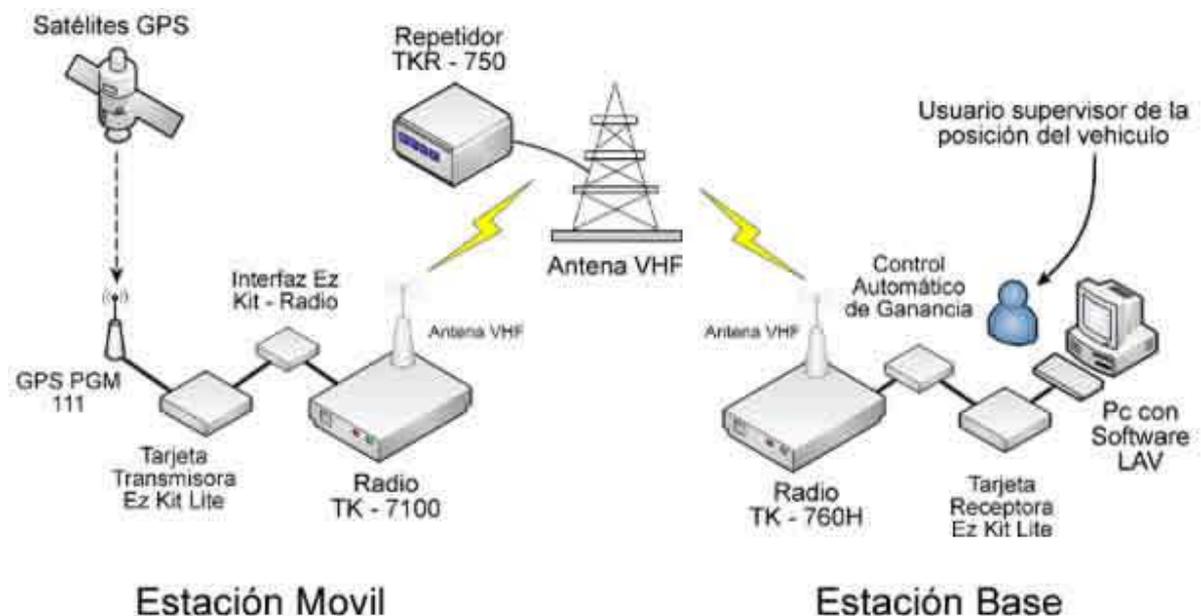


Figura 4. 25: Esquema General de la pruebas de campo

Estas pruebas son realizadas utilizando la repetidora instalada, para lo cual se configura las radio bases transmisora y receptora para que operen en este modo, algunas evaluaciones son realizadas con o sin la intervención de la etapa de software LAV y son precisamente estas últimas las que son descritas en el presente numeral, los resultados serán visualizados entonces en un software terminal.

#### **4.2.2.1. Primera Prueba**

La primera prueba realizada en este modo consistió en una caminata en las inmediaciones del centro de desarrollo portando la primera versión del prototipo transmisor, mientras que el receptor aguarda en la central esperando recibir las tramas enviadas, mostrándoselas a través del software terminal.

Los objetivos de esta prueba fueron:

- Verificar el correcto funcionamiento del sistema de radio VHF en modo semi dúplex a través de una prueba de calidad de voz, evaluándola en base a una escala de diez.
- Comprobar el desenvolvimiento de los algoritmos de transmisión y recepción en desarrollo.
- Confirmar la precisión del dispositivo GPS usado en la obtención de la posición, cotejando las posiciones reales ocupadas en la caminata con las obtenidas por el receptor GPS.

Los resultados obtenidos fueron:

- El sistema de radio VHF se encuentra en buenas condiciones, obteniendo en la evaluación de calidad de voz una calificación de ocho por cuanto en posiciones cercanas al centro de desarrollo se tiene interferencia de ruido, posiblemente por la cercanía entre las antenas transmisora y receptora.

- Los resultados en cuanto al envío y recepción de tramas de localización se pueden visualizar en la figura 4.19 que corresponde a una ventana de terminal perteneciente a la documentación de esta prueba.

Como los datos de localización están en un formato poco usual de entender se los muestra a continuación graficados en el google earth, donde puede comprobarse de forma clara la ruta tomada en la prueba.



**Figura 4. 26:** Ruta seguida en la primera prueba

- La gráfica anterior muestra la mayoría de las posiciones ocupadas en la ruta seguida, exceptuando la número seis que esta fuera de misma, por tanto se puede concluir que los datos de localización provistos por el GPS son muy aproximadamente exactos pero no están exentos de fallas eventuales.
- La prueba tuvo éxito ya que el cien por ciento de tramas enviadas fueron recibidas sin novedad, se comprobó entonces el correcto funcionamiento del

sistema VHF y de los algoritmos que hasta aquel momento se encontraban operativos.

#### 4.2.2.2. Segunda Prueba

La segunda prueba realizada en este modo hizo uso de un vehículo en el cual se montó la primera versión del prototipo transmisor, se circuló en un área mayor que en la primera prueba, solamente la radio base fue conectada a la batería del auto, mientras que la tarjeta de desarrollo se alimenta mediante la batería acoplada; de igual manera los resultados se obtienen en el software terminal.

Los objetivos de esta segunda evaluación fueron:

- Determinar la operación de los prototipos en condiciones más reales que la anterior prueba.
- Descartar un problema en el sistema VHF recorriendo una zona mayor a la primera confirmando o no que el problema hallado en la prueba uno se debe a la cercanía de las antenas transmisora y receptora.



**Figura 4. 27:** Prototipo transmisor y auto en prueba dos

#### Resultados obtenidos:

- Hubo un problema en el funcionamiento del prototipo, ya que las referencias circuitales de alimentación eran diferentes para la tarjeta de desarrollo y para la radio base, provocando que el kit de desarrollo dejara de transmitir la información de localización, esto fue solucionado desconectando la radio base de la batería del auto y conectándola a la batería acoplada.
- Se comprobó que efectivamente el problema de interferencia de ruido se induce cuando las antenas transmisora y receptora se encuentran muy cerca, lo cual se concluyó ya que este solamente ocurrió en el punto uno, señalado en la figura 4.29, en los demás puntos mostrados no hubo este evento, sin embargo a pesar de ello, se recibió correctamente la trama correspondiente al punto uno.
- Los resultados en cuanto al envío y recepción de tramas de navegación se pueden visualizar en la figura 4.28 que corresponde a una ventana de terminal perteneciente a la documentación de esta prueba.
- Salvo los problemas físicos mencionados, el sistema tuvo cien por ciento de éxito en la prueba habiendo recibido el total de tramas enviadas. Se comprobó entonces el correcto funcionamiento del sistema VHF y de los algoritmos que hasta aquel momento se encontraban operativos.

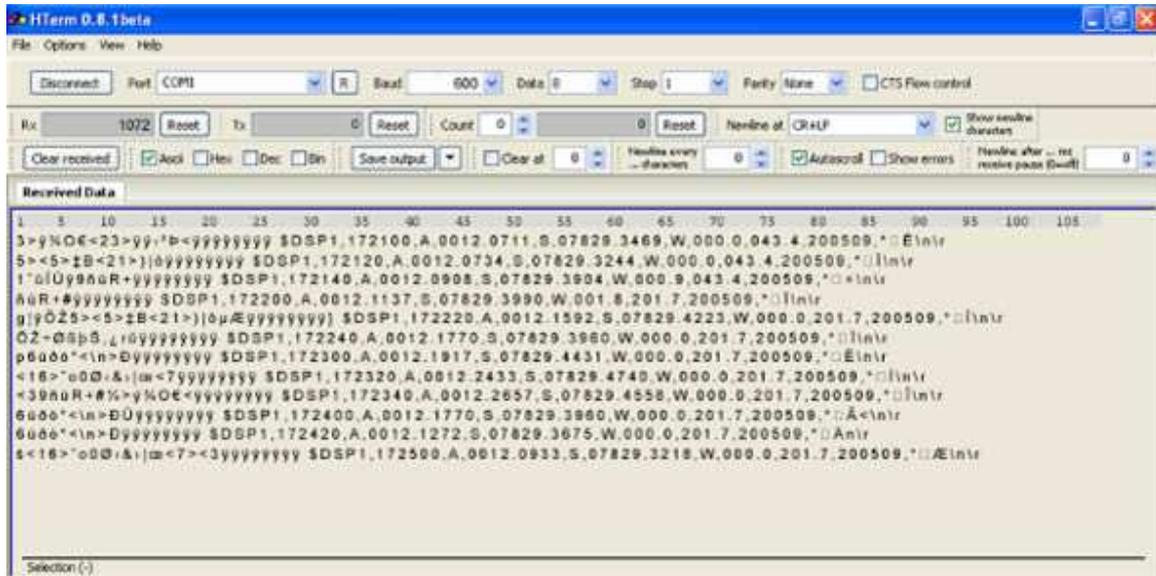


Figura 4. 28: Resultados Prueba dos

La información geográfica se pasa al google earth para mayor entendimiento:

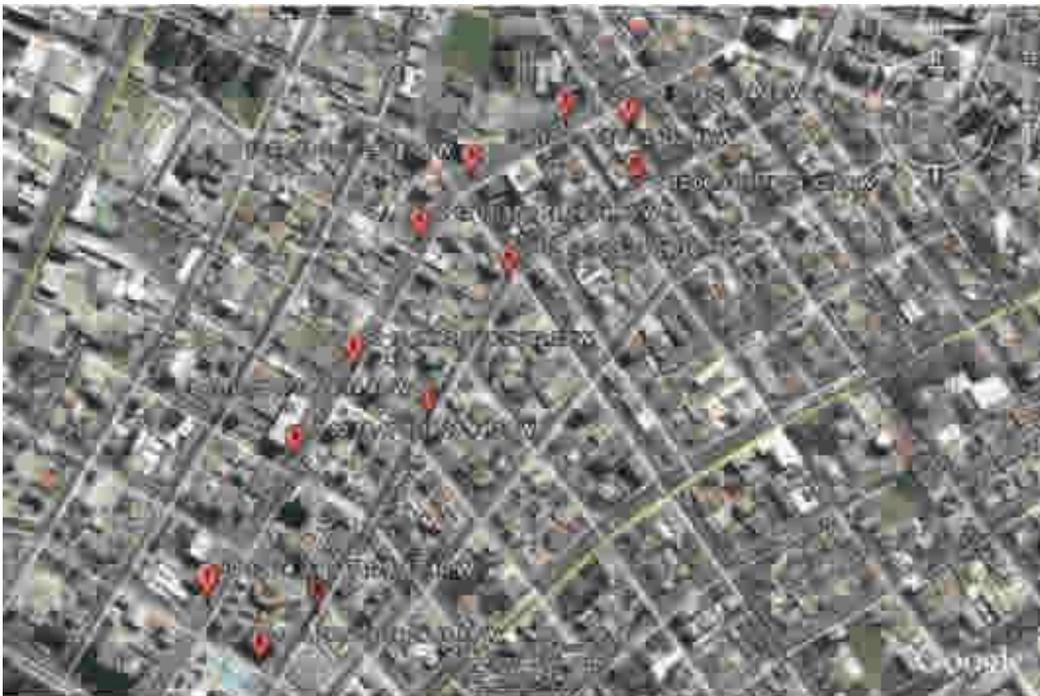


Figura 4. 29: Ruta seguida en la prueba dos

#### 4.2.2.3. Tercera Prueba

La tercera prueba consistió en una caminata en las proximidades del centro de desarrollo, dicha evaluación se la hace con las segundas versiones de los prototipos transmisor y receptor.

Objetivos de la prueba:

- Probar la segunda versión de la interfaz tarjeta de desarrollo – radio base descrita en el numeral 4.1.2<sup>83</sup>. figura 4.5. del presente capítulo
- Probar el circuito de control automático de ganancia variando el volumen de forma aleatoria en recepción.
- Evaluar los nuevos algoritmos de discriminación inteligente de bits y validación de tramas recibidas, en una operación semejante a la real.
- Verificar el nuevo formato de trama que incluye caracteres de inicio (>) y fin de la misma (<) utilizados en la etapa de software.

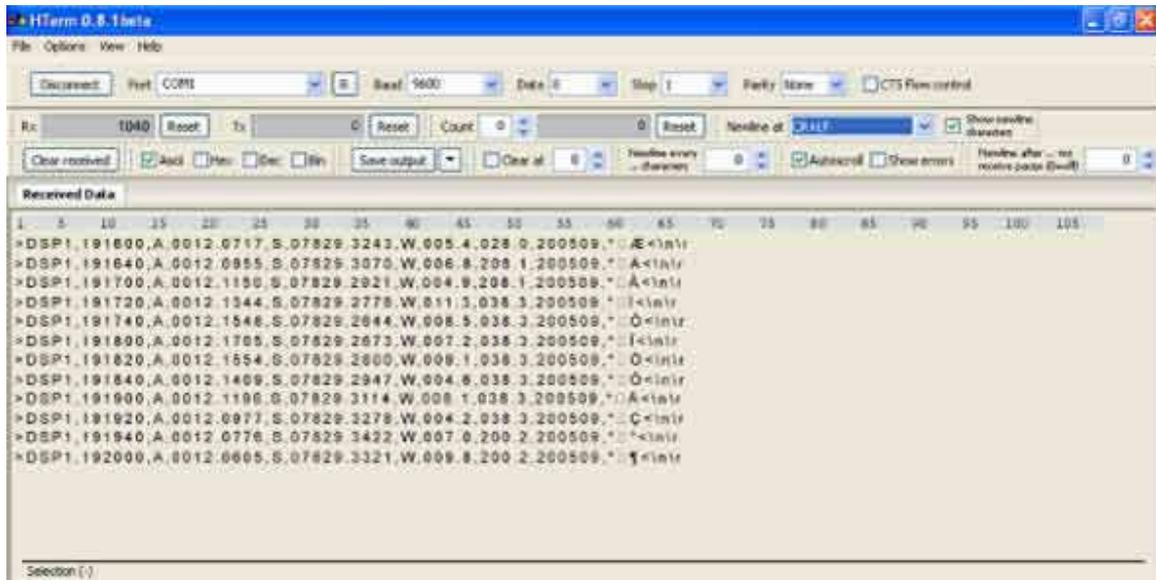
Resultados obtenidos:

- La segunda versión de la interfaz tarjeta de desarrollo – radio base funciona correctamente, pero la necesidad de permitir una comunicación de voz es evidente por lo que se decide reemplazarla de manera inmediata.
- El circuito de control automático de ganancia funciona correctamente y de forma igual a lo presentado en el numeral 4.2.1.2.1<sup>84</sup>.
- Los nuevos algoritmos operaron de la manera esperada, cuyos resultados pueden visualizarse en la ventana terminal mostrada a continuación:

---

<sup>83</sup> Página 184 del presente proyecto de titulación

<sup>84</sup> Página 200 del presente proyecto de titulación



**Figura 4. 30:** Resultados prueba tres

La mejora respecto a otras pruebas es evidente, las tramas ahora son validadas y presentadas en forma limpia sin el molesto ruido del canal.

- En la misma figura puede observarse los caracteres añadidos de inicio y fin de trama.
- Se obtuvo un cien por ciento de tramas recibidas por lo que se comprobó entonces el correcto funcionamiento del sistema VHF y de los algoritmos que hasta aquel momento se encontraban operativos.

La ruta de la evaluación se la observa a continuación:



**Figura 4. 31:** Ruta de la prueba tercera

#### 4.2.2.4. Cuarta Prueba

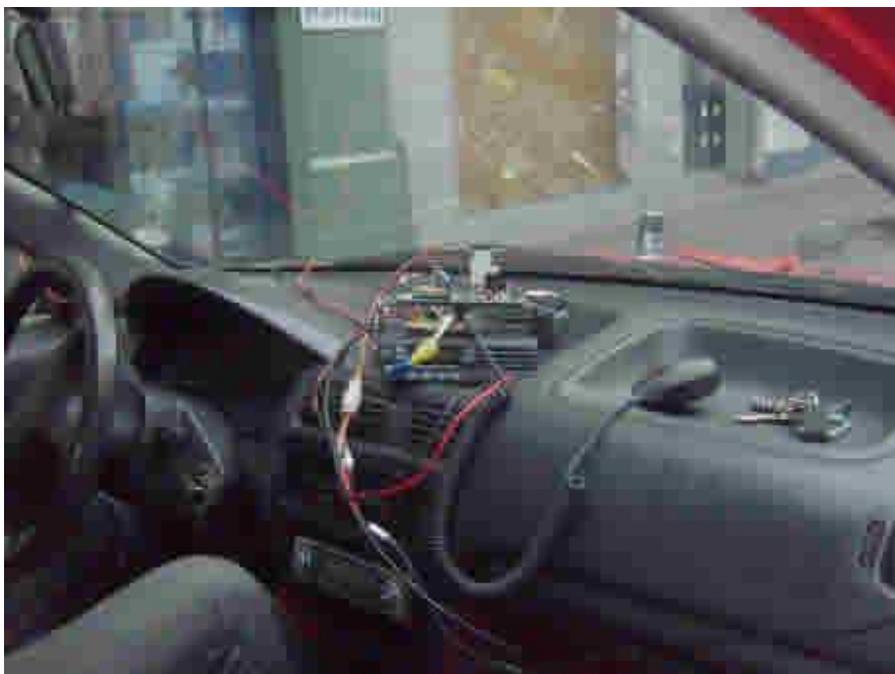
La cuarta prueba se realizó con la tercera versión del prototipo transmisor, el cual se montó en un vehículo diferente al de la prueba segunda, en esta evaluación se corrigió el problema de las referencias circuitales, pues, ahora tanto la radio base como el kit de desarrollo se encuentran conectados a la batería del auto.

Objetivos de la prueba:

- Verificar el funcionamiento del prototipo con la corrección aludida en el párrafo anterior.
- Verificar el funcionamiento de la nueva interfaz tarjeta de desarrollo – radio base, que permite físicamente la coexistencia entre voz e información de localización y su algoritmo relacionado (*módulo de apertura automática del canal*).
- Verificar el funcionamiento del estado de pánico.
- Verificar en primera instancia la cobertura del sistema VHF.



**Figura 4. 32:** Prototipo transmisor y auto en prueba cuatro



**Figura 4. 33:** Entorno de prueba cuatro

Los resultados obtenidos fueron:

- La corrección realizada en el prototipo para usar una misma fuente de alimentación para el kit de desarrollo y la radio base tuvo éxito.
- El circuito interfaz operó tal como se lo esperaba, sin embargo se presenta la necesidad de realizar una modificación que permita dar prioridad a las comunicaciones por voz sin que se pierda el turno para transmitir datos.
- Las transmisiones de información por estado de pánico funcionan correctamente, obteniéndose las tramas de posición cada intervalo de tiempo de veinte segundos, permaneciendo en este estado todo el tiempo que duró la prueba.
- Se recorrió una mayor área en la presente prueba que en las anteriores, lo que será visualizado en la figura 4.33, la ventana de terminal correspondiente, tiene un contenido similar al mostrado en la prueba tercera.



**Figura 4. 34:** Ruta recorrida prueba cuatro

- En cuanto a la recepción de tramas, se tuvo una eficiencia del 95.71 por ciento ya que de un total de 350 tramas llegaron 335, las quince que no fueron recibidas se perdieron debido a la naturaleza cambiante del canal inalámbrico de comunicaciones debido a la influencia de obstáculos en el recorrido, se comprobó entonces el correcto funcionamiento del sistema VHF y de los algoritmos implementados.

#### **4.2.2.5. Quinta Prueba**

La quinta prueba se realizó con las versiones definitivas de los prototipos transmisor y receptor y con las adecuaciones mencionadas en la prueba anterior en cuanto a la convivencia de voz y datos, además se implementó la transmisión de la información a 1200 bps; dicha prueba se realizó en un auto diferente a los de anteriores evaluaciones.

Objetivos de la prueba:

- Comprobar el rendimiento total del sistema con la nueva velocidad de transmisión de datos.
- Verificar el funcionamiento del módulo de apertura automática del canal en su versión final, el cual permite la convivencia con las comunicaciones por voz, modificando para esto hardware y firmware para dar prioridad a voz y sin perder el turno de envío de información de localización.
- Determinar los límites del área de cobertura.



**Figura 4. 35:** Prototipo transmisor y auto en prueba cinco



**Figura 4. 36:** Entorno de prueba cinco

Resultados obtenidos:

- A 1200 bps el sistema funcionó de manera óptima sin inconvenientes.
- El módulo final de apertura automática del canal funcionó de la mera esperada.
- Los límites de cobertura se señalan mediante la siguiente ilustración en google earth:



**Figura 4. 37:** Límites del área de cobertura

#### 4.2.2.6. Sexta Prueba

Como lo indica la figura 4.6 estas prueba es realizada para evaluar la visualización de la posición en tiempo real del vehículo en el google earth y verificar el almacenamiento posicional en la base de datos de Microsoft Access, la prueba cinco tuvo como objetivo el hallar los límites aproximados del área de cobertura, por otro

lado en esta prueba se hace un recorrido minucioso entre los límites arrojados por la prueba cinco obteniendo así un mapa aproximado de cobertura.

Objetivos de la prueba:

- Realizar un mapa de cobertura lo más aproximado posible, donde se puedan observar zonas en que la señal es óptima para la localización vehicular, zonas en donde la señal es débil, donde podría o no ubicarse al móvil y lugares donde no es posible realizar localización vehicular.
- Comprobar el funcionamiento de la parte de software del sistema LAV implementado, para lo cual se observa al móvil en tiempo real en el software SIG Google Earth al tiempo que se guardan los datos de navegación en la base de datos de Access.
- Comprobar una vez más la confiabilidad de la plataforma de datos implementada para el transporte de información de navegación.

Resultados Obtenidos:

- La parte de software funcionó de acuerdo a lo esperado, pudiendo observarse la posición del móvil en tiempo real y posibilitando la elaboración del mapa de cobertura que más adelante se describe. Del mismo modo la base de Datos guardó todos los elementos transmitidos por la unidad móvil.
- El sistema finalmente implementado funcionó cualitativamente de acuerdo a lo esperado del mismo.
- Es difícil hablar de resultados cuantitativos en esta prueba por cuanto el móvil realizó recorridos tanto en áreas de cobertura alta, media, baja y nula, en las cuales hubo pérdidas de información de navegación lo cual sirvió para realizar el mapa de cobertura mostrado a continuación:

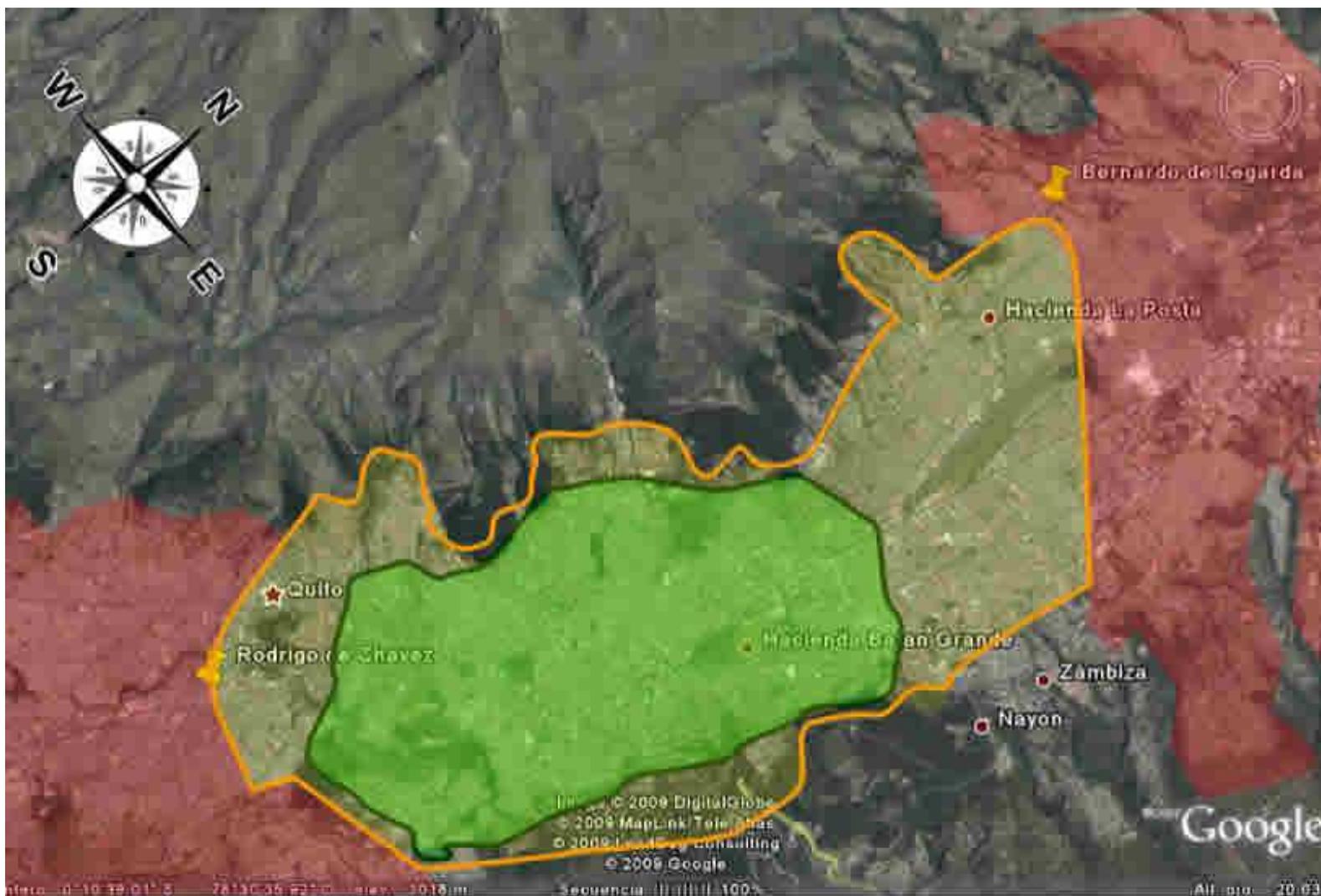


Figura 4. 38: Mapa de cobertura del sistema LAV

Se tiene la siguiente nomenclatura para la figura anterior:

**Nivel de Señal  
óptimo**

Los límites geográficos aproximados para esta área son:

Norte: Av. Galo Plaza Lasso, Sector El Labrador, Av. Edmundo Carvajal

Sur: Av. Maldonado Terminal Terrestre Cumandá, Av. Vicente Rocafuerte, Sector La merced 2

Oriente: Av. Oriental

Occidente: Av. Occidental

**Nivel de Señal débil**

Los límites geográficos aproximados para esta área son:

Norte: Av. Bernardo de Legarda, Sector San Carlos, Av. Ramón Chiriboga, Sector La Ofelia, Av. Eloy Alfaro Sector Fátima 1.

Sur: Av. Rodrigo de Chávez, sectores La Villaflora, La León, Las 5 Esquinas.

Oriente: Autopista Gral. Rumiñahui (Monjas)

Occidente: Av. Occidental

**Nivel de Señal nulo**

Esta área constituye las afueras de lo delimitado por la sección amarilla.

La ciudad de Quito se encuentra asentada en una zona geográfica no regular, existiendo planicies, depresiones y elevaciones, todo lo cual complica la realización de un mapa de cobertura casi exacto, el realizado en esta prueba pretende ser solamente una aproximación.

## **CAPÍTULO 5**

### **CONCLUSIONES Y RECOMENDACIONES**

A través del lapso de planificación, desarrollo y finalización del presente proyecto de titulación se pudieron destacar las siguientes conclusiones y recomendaciones.

#### **5.1. CONCLUSIONES**

- Se ha cumplido con el objetivo general del presente proyecto de titulación a través de sus objetivos específicos, tomando en cuenta, e inclusive superando el alcance del mismo, por cuanto inicialmente se planteó visualizar el móvil en tiempo real en el google earth, a través del envío de información de posicionamiento vía radio VHF, sin embargo no se mencionó un satisfactorio acoplamiento con la parte de voz y la inclusión de un procedimiento sencillo de evaluación de errores, lo cual se lo ha hecho estableciendo un sistema integral de voz – localización.
- Los resultados satisfactorios obtenidos en las pruebas finales de campo demuestran en la práctica la factibilidad de implementar el presente proyecto con una visión comercial.
- Un proyecto en desarrollo siempre permanece en continuo mejoramiento, inclusive si este se lo ha dado por satisfactoriamente finalizado, tal es nuestro caso en el que varias ideas de mejora quedaron en nuestra mente, las cuales pueden ser vistas como un trabajo futuro.
- El uso de comunicaciones por voz mediante sistema de cobertura VHF son ampliamente usados, por lo que la integración de un servicio de localización vehicular resultaría favorable y muy interesante, por cuanto, a parte de las ya mencionadas ventajas de dicho servicio, habría notables ahorros ya que se hace uso de una infraestructura instalada y un radio enlace probado.

- Los requerimientos del usuario delimitan la aplicabilidad de cualquier sistema LAV, pudiendo desarrollarse soluciones particulares dedicadas, la solución planteada para el presente proyecto de titulación considera una flota vehicular de máximo cincuenta vehículos, los cuales en teoría pueden ser ubicados en máximo treinta segundos mediante la transmisión de información de navegación a una velocidad de 1200 bps.
- La inclusión de un sistema repetidor permitió aumentar considerablemente el área de cobertura posibilitando de esta manera la realización de pruebas de campo en un ambiente inalámbrico real, con obstáculos y posibles interferencias que pongan a prueba la robustez del sistema, obteniendo de esta manera excelentes resultados, como puede comprobarse en la respectiva sección.
- Quito al ser una ciudad andina presenta una geografía peculiar haciendo que el medio inalámbrico presente mayores irregularidades, provocando de esta manera discontinuidades en las zonas de cobertura y un ligero aumento del error en la recepción de tramas de navegación en dichas zonas, sin embargo la robustez de los algoritmos implementados y la buena calidad del radioenlace permitió en la mayoría de los casos obtener resultados satisfactorios.
- Se ha develado una vez más las ventajas que proporciona el procesamiento digital de señal y su importancia en el desarrollo de aplicaciones prácticas como la que ha tratado el presente proyecto de titulación, claro está, a través del uso de un procesador digital de señal (*DSP*), una de las principales ventajas es la reprogramabilidad, así es perfectamente viable una configuración hardware que pueda ser programada para realizar una amplia variedad de tareas de proceso de señales, simplemente cambiándole el programa de funcionamiento. Por ejemplo, un filtro digital puede pasar de funcionar como pasa bajo a pasa alto sin necesidad de tocar para nada el

hardware. En un sistema analógico sería necesario cambiar el diseño completamente.

- Mediante el procesamiento digital de señal se pueden realizar diversas aplicaciones en ingeniería debido a las capacidades únicas de un DSP, el ejemplo clásico son los filtros en tiempo real de fase lineal, que de otro modo no son implementables.
- Un Procesador digital de señal es entonces un dispositivo con capacidad de procesamiento en línea, o en tiempo real, su arquitectura presenta unidades computacionales de gran potencia de cálculo preparadas para el tratamiento digital de señales, realizando el mayor número de operaciones aritméticas en el menor tiempo posible, para lo cual su circuitería ha sido optimizada.
- El Ez Kit Lite ADSP 21-81 constituye una herramienta interesantísima, la cual puede ser usada para desarrollar cualquier aplicación de procesamiento digital de señal y como tal no ha perdido vigencia, sin embargo en la actualidad existen instrumentos similares a este kit de desarrollo pero de mayor capacidad de procesamiento e incluso a un menor costo, así por ejemplo solo el microcontrolador DSP 2181 tiene un precio actual unitario de 49.4 dólares<sup>85</sup>, mientras que un DSP 2191 el valor unitario es de 32.5 dólares<sup>86</sup>.
- Se ha verificado en la práctica, las características de las diferentes técnicas de modulación pertenecientes a radio digital y se ha determinado la idoneidad de cada una para formar parte del sistema procesador de información.
- En el desarrollo de un sistema transmisor de datos es importante el ancho de banda del cual se disponga por cuanto de este depende la máxima velocidad de señal que se pueda lograr, es este entonces un recurso limitante que dependiendo de los requerimientos del usuario puede ser aun más explotado

---

<sup>85</sup> <http://search.digikey.com/scripts/DkSearch/dksus.dll?Detail&name=ADSP-2181KSTZ-133-ND>

<sup>86</sup> <http://search.digikey.com/scripts/DkSearch/dksus.dll?Detail&name=ADSP-2191MKSTZ-160-ND>

aplicando métodos de modulación diferentes de los que aquí se ha tratado, por ejemplo esquemas multinivel.

- El tipo de modulación digital que se utiliza en un sistema de telecomunicaciones, puede ser optimizado dependiendo del algoritmo de demodulación que se utilice, es decir, para cada técnica de modulación, pueden o no existir varias técnicas o algoritmos de demodulación que le correspondan.
- El uso de lenguaje assembler tiene como consecuencia que el código fuente sea no portable, debido a la dependencia de la arquitectura correspondiente, sin embargo este no resulta ser tan complejo y es por contrario de fácil e inmediata comprensión.

## **5.2. RECOMENDACIONES**

- El prototipo de sistema de localización vía radio VHF es una muestra de las potencialidades en la reutilización, y por tanto revalorización, de la infraestructura instalada, sin embargo para una implementación comercial es necesario realizar modificaciones en el aspecto de hardware, así para implementar el transmisor puede utilizarse un microcontrolador de bajo costo, mientras que para el receptor es necesario un procesador digital de señal (DSP) por las estructuras digitales de tiempo real que integra, por ejemplo los filtros digitales FIR y que no pueden realizarse eficientemente de otra manera.
- Los tiempos tomados para la transmisión de la información de navegación, sea esta por movimiento, reposo o pánico, son tomados en base a criterios empíricos, sería necesario entonces determinarlos en base a un estudio que tome en consideración la frecuencia de uso del canal por transmisiones de voz, el número de móviles a localizar y la base de tiempo para ubicar a todos los móviles.

- Siempre que se quiera implementar un algoritmo en cualquier microprocesador, se debe realizar previamente un diagrama de flujo, en el que consten todos aquellos procedimientos que se desean efectuar, para que de esta manera, el código pueda ser portable, es decir pueda implementarse en arquitecturas diferentes, para que el mismo pueda incluir modificaciones sin necesidad de reescribirlo completamente y sobre todo para que sea más entendible.
- Al configurar el modo de operación semi – dúplex es fundamental recortar las antenas de transmisión y recepción con a la longitud adecuada a su respectiva frecuencia de operación, para evitar problemas en la comunicación y daños en los equipos debido a un exceso de potencia reflejada.
- Así mismo en la configuración del sistema de radio es fundamental realizar un correcto acoplamiento de las antenas con su respectivo equipo de tal modo que se evite la intromisión de ruido que puede afectar la calidad del enlace y afecte tanto a las comunicaciones de voz como de datos.
- En la operación semi – dúplex no se deben acercar demasiado las antenas ya que se produce un efecto de retroalimentación produciéndose errores, se debe entonces mantener una distancia mínima de 10 metros.
- El presente trabajo podría alcanzar un mayor nivel, reemplazando la parte de radio frecuencia (radio bases), con un sistema que incluya un sintetizador digital directo, el que puede generar señales de cualquier frecuencia, con lo cual se puede modular directamente las señales que representan la información de navegación.
- El sistema desarrollado en el presente proyecto de titulación realiza un exitoso acoplamiento entre las partes datos y de voz, no hallándose interferencia alguna entre las mismas a pesar de funcionar en un mismo dispositivo transmisor, sin embargo sería necesario realizar un control de flujo para

identificar los momentos en los cuales el canal se halla ocupado, debido a envíos de otros transmisores, para así determinar los instantes precisos de transmisión y evitar pérdida de datos.

- Así mismo el sistema puede ser completado incluyendo bi - direccionalidad en la comunicación de datos, así por ejemplo se puede usar un procesador digital de señal para implementar un modulador – demodulador en los móviles, que aparte de realizar las tareas ya explicadas a lo largo del presente escrito, pueda recibir y procesar señales provenientes de una central permitiendo el manejo de actuadores acoplados al sistema, de modo que se obtenga un mayor control del automóvil en casos en que se lo requiera.
- Como se mencionó en la sección de conclusiones, acerca de la geografía de la ciudad de Quito, esta no permite que se tenga una cobertura total de la misma, debiendo ubicar el repetidor en un sitio más estratégico e inclusive situando otros.
- Para un sistema de localización vehicular más exigente, por ejemplo con una cantidad mayor de vehículos, se puede mejorar la eficiencia en el uso del ancho de banda disponible usando técnicas de modulación multinivel para mejorar la velocidad de transmisión.

## **BIBLIOGRAFÍA:**

- ANALOG DEVICES, adsp – 21xx Applications Handbook, Volumen 2, Analog Devices, USA, 1995.
- ANALOG DEVICES, adsp – 2100 family user's manual, Tercera edición, Analog Devices, USA, 1995.
- Peter and Joe, NMEA and GPS info page
- TOMASI, Wayne, Sistemas de Comunicaciones Electrónicas, Cuarta Edición, Pearson, México, 2003.
- VASQUEZ, Fernando, Implementación de un laboratorio básico de Telecomunicaciones, Escuela Politécnica Nacional, 1997.