



# **Escuela Superior Politécnica de Chimborazo**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

**ESCUELA DE INGENIERÍA EN ELECTRÓNICA**

**“DISEÑO E IMPLEMENTACIÓN DE UN ROBOT  
DETECTOR DE OBSTÁCULOS Y NAVEGACIÓN GPS”**

**“Tesis de grado previa la obtención del TÍTULO DE  
INGENIERO EN ELECTRÓNICA Y COMPUTACION”**

**Presentado por:**

**Paúl Aguilar Albán**

**Geovanny Simba Machado**

**Riobamba 2009**

A nuestros Padres que con su colaboración y apoyo constantes fueron parte importante en el desarrollo del presente trabajo de tesis.

**Firma**

**Fecha**

Dr. Romeo Rodríguez .....  
DECANO DE LA FACULTAD DE  
INFORMATICA Y ELECTRONICA

Ing. Paúl Romero .....  
DIRECTOR DE LA ESCUELA DE  
INGENIERIA ELECTRONICA Y TECNOLOGIA  
EN COMPUTACION

Ing. Paul Romero .....  
DIRECTOR DE TESIS

Ing. Wladimir Castro .....  
MIEMBRO DE TRIBUNAL

Lcdo. Carlos Rodríguez .....  
DIRECTOR DEL CENTRO DE DOCUMENTACION

Nota .....

“Nosotros, Paul Aguilar Albán y Geovanny Simba Machado, somos los responsables de las ideas, doctrinas y resultados expuestos en esta Tesis y el patrimonio intelectual de la misma pertenecen a la Escuela Superior Politécnica de Chimborazo “

.....

Paúl Aguilar Albán

.....

Geovanny Simba Machado

# Índice General

## Capítulo I

<b>Robótica, sistemas de navegación y ultrasonido .....</b>	<b>18</b>
1.1. Robótica .....	18
1.1.1. Concepto de robótica .....	18
1.2. Localización .....	19
1.2.1. Algoritmos de localización .....	19
1.3. Navegación con el sistema de posicionamiento global .....	20
1.3.1. Coordenadas terrestres .....	20
1.3.2. Localización geográfica de un punto .....	22
1.4. El rumbo y la brújula .....	25
1.4.1. Sistema de medición .....	26
1.4.2. La declinación magnética .....	26
1.5. El sistema de posicionamiento global .....	27
1.5.1. Errores en el sistema GPS .....	31
1.5.2. Numero de satélites visibles .....	31
1.5.3. Desviación por los relojes internos de los satélites .....	32
1.5.4. Errores orbitales .....	32
1.6. El sistema WGS84 .....	32
1.7. El estándar NMEA 0183 .....	34
1.7.1. Interfaz eléctrica .....	35
1.7.2. El formato general de las sentencias .....	36
1.8. Ultrasonido .....	43
1.8.1. Características .....	44
1.8.2. Fuentes de error en el manejo de ultrasonido .....	46
1.8.3. Sensores de ultrasonido .....	49

## Capítulo II

<b>Diseño mecánico .....</b>	<b>50</b>
2.1. Modos de locomoción .....	50
2.1.1. Ruedas de direccionamiento .....	51

2.1.2. Modelo de triciclo .....	51
2.1.3. Locomoción de robots móviles .....	52
2.1.3. Plataforma de locomoción .....	55
2.2. Sistemas de tracción .....	55
2.3. Diseño de la estructura .....	59

## Capítulo III

<b>Diseño electrónico .....</b>	<b>61</b>
3.1. Esquema general del sistema .....	61
3.2. Fuente de alimentación .....	62
3.2.1. Módulo regulador y distribuidor de voltaje .....	63
3.3. Circuito interruptor digital .....	65
3.4. Adquisición del entorno por ultrasonido .....	66
3.4.1. Sensor ultrasónico Max Sonar EZ1 .....	66
3.5. Diseño del módulo ultrasónico detector de obstáculos .....	71
3.6. Etapa de control de motores .....	73
3.6.1. Driver de potencia tipo puente H .....	74
3.6.2. Control de velocidad PWM .....	78
3.7. La brújula digital CMPS03 .....	82
3.7.1. Conexiones y funcionamiento .....	82
3.7.2. Registros de la brújula digital CMPS03 .....	84
3.8. Módulo receptor GPS SPK-GPS-GS405 .....	85
3.8.1. Características principales .....	86
3.8.2. Especificaciones técnicas y eléctricas .....	86
3.8.3. Descripción de los pines .....	88
3.9. Tarjeta principal .....	89
3.9.1. Control de potencia para motores DC .....	89
3.9.2. Sensores ultrasónicos .....	89
3.9.3. Interfaz humana .....	89
3.10. Microprocesador para el cálculo matemático .....	91
3.10.1. Características del núcleo del micro controlador .....	94
3.10.2. Características de los periféricos .....	95

## Capítulo IV

<b>Software.....</b>	<b>98</b>
4.1. Metodología .....	98
4.2. Análisis.....	99
4.2.1. Requisitos.....	100
4.3. Diseño.....	102
4.3.1. Control de teclado .....	102
4.3.2. Comunicación con el modulo receptor GPS .....	104
4.3.3. Control modulo sonar .....	106
4.3.4. Algoritmo de promedio.....	107
4.3.5. Comunicación del modulo sonar .....	109
4.3.6. Comunicación de la brújula electrónica.....	111
4.4. Integración del programa principal .....	112
4.4.1. Navegación por satélite .....	112
4.4.2. Comunicación con el PIC matemático .....	114
4.4.3. Programa de evasión de obstáculos .....	115
4.5. Calculo del rumbo.....	117
4.5.1. Calculo del vector fuerza repulsiva .....	121
4.5.2. Calculo del vector fuerza resultante .....	122
4.5.3. Control de movimiento por PWM .....	124
4.6. Implementación .....	127
4.6.1. Comunicación con el módulo receptor GPS .....	127
4.6.2. Comunicación con el módulo sonar.....	128
4.6.3. Comunicación con la brújula electrónica.....	129
4.6.4. Comunicación con e PIC matemático .....	130
4.6.5. Programa de evasión de obstáculos .....	131
4.7. Pruebas y validación.....	132

## Capitulo V

<b>Análisis de pruebas y resultados.....</b>	<b>133</b>
5.1. Prueba y análisis del modulo sonar .....	133
5.1.1. Prueba de los sensores ultrasónicos.....	133
5.1.2. Diseño de la prueba .....	134

5.1.3.Análisis de la ubicación de los sensores .....	136
5.2.Pruebas del algoritmo de navegación .....	138
5.2.1.Prueba del algoritmo de cálculo del rumbo.....	138
5.2.2.Diseño de la prueba .....	138
5.2.3.Resultados de la prueba.....	139
5.3.Pruebas del algoritmo de control de giro y velocidad.....	140
5.3.1.Diseño de la prueba .....	141
5.3.2.Resultados de la prueba.....	141
5.4.Prueba del algoritmo de navegación GPS sin obstáculos.....	142
5.4.1.Diseño de la prueba .....	142
5.4.2.Resultados de la prueba.....	144
5.5.Prueba del algoritmo de navegación GPS con un obstáculo.....	145
5.5.1.Diseño de la prueba .....	145
5.5.2.Resultados de la prueba.....	147
5.6.Prueba del algoritmo de navegación GPS con tres obstáculos .....	147
5.6.1.Diseño de la prueba .....	148
5.6.2.Resultados de la prueba.....	149
5.7.Prueba del algoritmo de navegación GPS con seis obstáculos .....	150
5.7.1.Diseño de la prueba .....	152
5.7.2.Resultados de la prueba.....	152
5.8.Análisis de resultados.....	153

## Índice de ilustraciones

### Capítulo I

#### Robótica, sistemas de navegación y ultrasonido

Figura I.1. Meridianos y paralelos de la esfera planetaria .....	21
Figura I.2. Representación del sistema tridimensional.....	22
Figura I.3. Definición de longitud .....	23
Figura I.4. Definición de latitud .....	24
Figura I.5. Declinación de dos puntos diferentes en la Tierra.....	25
Figura I.6. La declinación magnética y el rumbo verdadero .....	25
Figura I.7. El rumbo en el sistema circular .....	26
Figura I.8. Los diferentes nortes y sus variaciones.....	27
Figura I.9. Satélite GPS .....	28
Figura I.10. Los tres segmentos del GPS .....	29
Figura I.11. La constelación de satélites de GPS que circundan la Tierra en orbitas muy precisas .....	31
Figura I.12. Errores de GPS que muestran una señal bloqueada y una señal multi – camino ...	32
Figura I.13. Ejes del sistema WGS 84 .....	33
Figura I.14. Distribución de las zonas de monitoreo GPS .....	34
Figura I.15. Reconocimiento por emisión recepción .....	44
Figura I.16. Incertidumbre angular en la medida de un ultrasonido .....	46
Figura I.17. Distancia mínima de reconocimiento .....	47
Figura I.18. El sensor “a” emite el pulso que recibe el sensor “b” .....	48
Figura I.19. Funcionamiento básico de un sensor ultrasónico .....	49

### Capítulo II

#### Diseño mecánico

Figura II.1. Ruedas de direccionamiento.....	51
Figura II.2. Centro instantáneo de rotación b) centro instantáneo de curvatura.....	52
Figura II.3. Robot con 0 grados de libertad.....	52
Figura II.4. Robot con 1 grados de libertad.....	53
Figura II.5. Robot con 2 grados de libertad.....	53
Figura II.6. Robot con 3 grados de libertad.....	53

Figura II.7. Robot con 0 grados de direccionamiento .....	54
Figura II.8. Robot con 1 grados de direccionamiento .....	54
Figura II.9. Robot con 2 grados de direccionamiento .....	54
Figura II.10. Sistema de tracción mediante engranajes.....	56
Figura II.11. Tren de engranajes usados en el robot.....	56
Figura II.12. Sistema de tracción .....	58
Figura II.13. Material de aluminio de 2 pulgadas para la estructura .....	59
Figura II.14. Construcción del chasis del robot .....	60
Figura II.15. Estructura de aluminio del robot .....	60

### **Capítulo III**

#### **Diseño electrónico**

Figura III.1. Diagrama de bloques del sistema robótico.....	62
Figura III.2. Batería de 12 Voltios a 7 amperios .....	63
Figura III.3. Placa de alimentación de múltiples salidas.....	63
Figura III.4. Diagrama del circuito electrónico para la tarjeta de alimentación.....	64
Figura III.5. Módulo del interruptor digital .....	65
Figura III.6. Diagrama del circuito electrónico para el interruptor electrónico .....	66
Figura III.7. Sensores de ultrasonido Max sonar EZ1 usados en el módulo sonar .....	67
Figura III.8. Diagrama del circuito del sensor Max sonar EZ1 .....	68
Figura III.9. Capacidad del rango del sensor ultrasónico .....	70
Figura III.10. Circuito de adquisición de datos para los sensores ultrasónicos.....	71
Figura III.11. Controlador de motores DC de potencia .....	74
Figura III.12. Configuración básica de un puente H .....	75
Figura III.13. Puente H con transistores Darlington .....	76
Figura III.14. Puente H con dos entradas de control.....	77
Figura III.15. Circuito equivalente de un motor DC.....	78
Figura III.16. Motor DC de alta inductancia b) motor DC de baja inductancia .....	79
Figura III.17. Esquema electrónico completo del puente H.....	81
Figura III.18. Brújula digital CMPS03 .....	82
Figura III.19. Pines de conexión de la brújula digital CMPS03 .....	83
Figura III.20. Funcionamiento del protocolo I2C.....	83
Figura III.21. Disposición de pines para el módulo GPS SPK GPS GS405.....	88

Figura III.22. Teclado matricial de 4x3 teclas .....	90
Figura III.23. Visualizador LCD de 16 caracteres por 2 líneas.....	91
Figura III.24. Diagrama del circuito de procesamiento matemático.....	93
Figura III.25. Diagrama electrónico de la tarjeta de control principal .....	97

## Capítulo IV

### Software

Figura IV.1. Metodología de desarrollo: modelo incremental.....	99
Figura IV.2. Nivel 0: DFD Diagrama de contexto del sistema.....	99
Figura IV.3. Nivel 1: DFD Diagrama del sistema robótico .....	101
Figura IV.4. Algoritmo de ingreso de coordenadas del objetivo.....	102
Figura IV.5. Algoritmo de prueba de sensores.....	103
Figura IV.6. Mensaje GGA de NMEA .....	104
Figura IV.7. Algoritmo de lectura de datos del modulo GPS.....	105
Figura IV.8. Tareas ejecutadas para el módulo sonar .....	107
Figura IV.9. Algoritmo de lectura y promedio.....	109
Figura IV.10. Algoritmo de lectura de datos para un sensor ultrasónico .....	110
Figura IV.11. Algoritmo de lectura de datos para la brújula electrónica .....	111
Figura IV.12. Secuencia principal para la navegación GPS.....	113
Figura IV.13. Algoritmos de comunicación para el PIC 116F877A y 18F452.....	115
Figura IV.14. Campo de fuerzas repulsivas generadas por la presencia de obstáculos en el área de trabajo .....	117
Figura IV.15. Determinación del rumbo.....	118
Figura IV.16. Triangulo formado por el punto de salida, de llegada y el polo norte .....	118
Figura IV.17. Algoritmo para el cálculo del vector fuerza de atracción .....	120
Figura IV.18. Algoritmo para el cálculo del vector fuerza repulsiva .....	122
Figura IV.19. Algoritmo para la determinación de la dirección del movimiento .....	123
Figura IV.20. Curvas de respuesta ente el ángulo $\theta_w$ y el control de velocidad PWM .....	126

## Capitulo V

### Análisis de pruebas y resultados

Figura V.1. Configuración inicial.....	136
--	-----

Figura V.2. Configuración final de los sensores ultrasónicos.....	137
Figura V.3.Navigator, calculadora del círculo máximo.....	139
Figura V.4. Entorno de la prueba sin obstáculos, destino punto B.....	143
Figura V.5. Entorno de la prueba sin obstáculos, destino punto A.....	144
Figura V.6. Entorno de la prueba con un obstáculo, destino punto B .....	146
Figura V.7. Entorno de la prueba con un obstáculo, destino punto A .....	146
Figura V.8. Entorno de la prueba con tres obstáculos, destino punto B .....	148
Figura V.9. Entorno de la prueba con tres obstáculos, destino punto A .....	149
Figura V.10. Entorno de la prueba con seis obstáculos, destino punto B .....	151
Figura V.11. Entorno de la prueba con seis obstáculos, destino punto A .....	152

## Índice de tablas

### Capítulo I

#### Robótica, sistemas de navegación y ultrasonido

Tabla I.1. Parámetros principales del sistema WGS 84.....	33
Tabla I.2.Parámetros de comunicación serial NMEA 0183 .....	35
Tabla I.3. Identificadores de un transmisor GPS .....	38
Tabla I.4.La variable GGA y su contenido.....	39
Tabla I.5. La variable GSA y su contenido.....	40
Tabla I.6. La variable GSV y su contenido.....	41
Tabla I. 7.Los valores RMC y su contenido .....	42
Tabla I.8. La variable GLL y su contenido .....	43
Tabla I.9. La variable VTG y su contenido .....	43

### Capítulo III

#### Diseño electrónico

Tabla III.1. Activación secuencial de sensores ultrasónicos .....	72
Tabla III.2. Control de la salida de datos de cada sensor hacia el micro controlador.....	73
Tabla III.3.Funcionamiento del puente H.....	77
Tabla III.4. Puente H modificado .....	78
Tabla III.5. Conversión de señales para la etapa de PWM .....	81
Tabla III.6. Registros internos de la brújula digital CMPS03 .....	85
Tabla III.7. Valores absolutos máximos de voltaje y corriente.....	86
Tabla III.8. Características de la antena integrada .....	87
Tabla III.9. Especificaciones técnicas.....	87
Tabla III.10. Especificaciones técnicas de precision .....	87
Tabla III.11. Varios modelos prospecto de PIC gama alta .....	92
Tabla III.12. Características técnicas de los PIC 18FXXX .....	92
Tabla III.13. Requerimientos de I/O de la tarjeta principal.....	93

## **Capítulo IV**

### **Software**

Tabla IV.1. Mensajes NMEA entregados por el módulo receptor GPS .....	104
Tabla IV.2. Promedio algebraico para 5 medidas realizadas por cada sensor .....	107
Tabla IV.3. Promedio aritmético revisado.....	108
Tabla IV.4. Parámetros de comunicación con el modulo sonar.....	110
Tabla IV. 5.Parámetros de comunicación con el modulo receptor GPS .....	127
Tabla IV. 6.Modelo incremental (incrementos desarrollados) .....	132

## **Capitulo V**

### **Análisis de pruebas y resultados**

Tabla V.1. Mediciones obtenidas para dos tipos de materiales .....	135
Tabla V. 2.Calculo del rumbo de navegación .....	140
Tabla V. 3.Relacion para los diferentes modos de giro.....	141
Tabla V. 4.Resultados de la prueba sin obstáculos .....	144
Tabla V. 5. Resultados de la prueba con un obstáculo .....	147
Tabla V. 6. Resultados de la prueba con tres obstáculos.....	150
Tabla V. 7. Resultados de la prueba con seis obstáculos .....	153

## **INTRODUCCIÓN**

El área de la robótica ha sido desde hace años fuente de grandes investigaciones. Gracias a estos trabajos, se han realizado avances en todas las tareas que incluye el concepto de hacer que un robot se mueva con autonomía en un área específica. Se han utilizado técnicas como el mapa de entorno, generación de trayectorias y caminos, detección de obstáculos y navegación global.

Sin embargo aún no existe una solución definitiva al problema de la robótica móvil, para cada entorno existe una solución específica que cambia según el grado de autonomía que se desee conceder, cada una con puntos a favor y en contra. Conforme se van generando nuevas soluciones, los algoritmos son cada vez más complejos y necesitan más carga computacional.

Los trabajos de navegación generalmente se basan en el mapa de algún entorno estructurado. Existen otros trabajos centrados en el posicionamiento del robot y reconocimiento del entorno que le rodea, mediante técnicas complejas basadas en distintos tipos de sensores y cámaras, sin embargo el algoritmo de control es demasiado complejo para ser ejecutado en tiempo real y con movimientos cómodos del robot.

Los sensores más utilizados para captar la información del entorno son de ultrasonidos, infrarrojos y visión artificial. Los dos primeros utilizados para la detección de obstáculos ya que permiten obtener una relación instantánea de distancias a objetos y el tercero tiene la capacidad de identificación de formas y colores pero con mayor requerimiento de procesamiento de datos.

Gracias a toda la información y referencias de investigaciones de robótica, la hipótesis que rige esta tesis asevera que es posible la implementación de un sistema de detección de obstáculos por sensores ultrasónicos y navegación GPS que permitirá mejorar las funciones de autonomía e interacción de un robot móvil en un campo de pruebas prediseñado.

Basados en lo anterior se busco alcanzar objetivos específicos como:

- Estudiar el funcionamiento de los sensores ultrasónicos aplicados a la robótica.
- Estudiar el funcionamiento del sistema de posicionamiento global (GPS).
- Diseñar la tarjeta de adquisición de señales de los sensores ultrasónicos para detección de obstáculos.
- Implementar un sistema de navegación por módulo GPS para la ubicación del destino.
- Diseñar e implementar la tarjeta de control de movimiento basado en motores DC.
- Diseñar e implementar el software de control del robot móvil.

Para lograrlo se siguió una metodología de investigación científica deductiva – experimental, cuya colaboración nos otorga obtención rápida de datos, experimentación a través de ensayo y error y los resultados obtenidos se basan en las variables reales encontradas en el campo de pruebas.

La elaboración del proyecto tuvo como sede un laboratorio electrónico instalado en el hogar donde se realizó la implementación previa la investigación de todos los antecedentes necesarios para iniciar con el trabajo. La investigación fue basada netamente en internet pues es una fuente increíblemente grande de información y donde se encontró además experiencias asociadas al tema en diferentes partes del mundo que ayudo a superar los inconvenientes más comunes.

Tras meses de trabajo, obtención y análisis de datos, se probaron los diferentes módulos que componen el proyecto para entender su modo de operación, determinar sus salidas y lo que necesitaba para trabajar sin

problemas. Se resolvieron problemas de alimentación, se depuró el software y se unificó todas las etapas para que integren un solo sistema a través de estándares y protocolos de comunicación serial, obteniendo datos del exterior, procesándolos y ejecutando ordenes que dirigen al móvil desde un punto inicial a uno final pre definido en un entorno de prueba pre diseñado.

El robot móvil es capaz de guiarse de forma autónoma a través de un entorno establecido con anterioridad librando obstáculos a su paso pero siempre dirigiéndose a un destino ingresado al sistema al inicio de la prueba y cuyas coordenadas dependen de su posición global dada por el modulo GPS que tiene instalado.

## **CAPITULO I**

### **ROBÓTICA, SISTEMAS DE NAVEGACIÓN Y ULTRASONIDO**

En este primer capítulo se presentan las generalidades acerca del concepto de robótica, sistema de posicionamiento global y sensores ultrasónicos. Estas generalidades permiten entender la visión global de todos los factores que intervienen y definirán de manera directa los alcances del trabajo aquí presentado.

#### **1.1. Robótica**

##### **1.1.1. Concepto de Robot**

De manera general un robot es una máquina con sensores, inteligencia y movilidad. Para que un sistema electromecánico sea calificado como un robot debe ser hábil para:

- Percibir y sensor el entorno que lo rodea
- Realizar tareas como locomoción o manipulación

- Ser re-programable para lograr múltiples tareas con el mismo equipo
- Ejecución autónoma e interacción con seres humanos

La sub-categoría de robótica a la cual pertenece el actual proyecto es la de robot de razonamiento, ya que convierte sus percepciones en acciones, específicamente en movimientos de locomoción guiados por el sistema de navegación.

Un sistema de navegación consiste en un conjunto de sensores, métodos y tecnologías que tratan de situar a un robot en su entorno, estos sistemas se clasifican en absolutos y relativos. Se usa un sistema de posicionamiento absoluto que no depende de un origen de referencia para poder ubicarse siempre en todo momento y reencontrarse si hubiese perdido la orientación

## **1.2. Localización**

La localización es uno de los problemas clásicos dentro del área de robótica en este caso, la robótica móvil. El problema consiste en que un robot se localiza automáticamente dentro de un determinado mundo usando para ello sus sensores y la información de su entorno que en general se ofrece en forma de un “mapa”, Cañas (4).

Dentro de la localización, se puede distinguir dos problemas distintos, el primero y el más sencillo es partir de una posición inicial conocida y el siguiente a partir de la localización global.

### **1.2.1. Algoritmos de localización**

Para resolver el problema de la localización hay que tener en cuenta varios parámetros que según cambia el entorno, cambia el problema y cambia la solución. Dicho esto el problema puede dividirse en “grados de dificultad”, se parte del más fácil que puede haber es la localización con “sensores de

posición”, en este caso la posición nos la dan los sensores de forma directa, como mucho habrá que corregir los errores.

Cuando se trata de localización en interiores los mejores candidatos son los encoders, este tipo de sensores se basa en calcular la diferencia entre lecturas sucesivas de las posiciones de las ruedas del robot, sin embargo cuando se trata de localizarse en exteriores, aparecen otros candidatos, pero el más utilizado es el GPS. Este sistema está basado en satélites girando alrededor de la tierra de manera continua y envían pulsos exactos de reloj.

Son necesarios al menos cuatro satélites para obtener las coordenadas de la posición y el tiempo. Se trata de un sistema que permite calcular las coordenadas de cualquier punto de la superficie terrestre. Su principal ventaja es que proporciona la localización absoluta en un área grande y sin requerir estructura alguna del entorno.

Las fuentes de error del GPS son distintas a las de los encoders, los errores no son acumulativos y pueden deberse a errores del reloj (máximo 1m.), errores en la órbita (máximo 1m.), errores en la modelización de la propagación en la troposfera y la ionosfera (1m. y 10m., respectivamente), errores debidos a rebotes de la señal (0.5m.) y errores de usuario en el receptor, Cañas (4).

### **1.3. Navegación con el sistema de posicionamiento global**

Antes de investigar la capacidad de una unidad de GPS, es vital entender los elementos esenciales del sistema GPS y los métodos de posicionamiento en la superficie terrestre.

#### **1.3.1. Coordenadas terrestres**

Todos los planos paralelos al plano ecuatorial cortan al elipsoide terrestre en círculos paralelos al ecuador que se llaman paralelos. Todos los círculos máximos que pasan por ambos polos Norte y Sur son perpendiculares al ecuador y a los paralelos, llamados estos círculos máximos meridianos, China (6).

Se consigue, de esta forma, "cuadricular" la superficie del planeta de forma que, si se numeran los paralelos y los meridianos se puede identificar cada uno de sus puntos. Es necesario, por tanto, fijar un origen tanto en la medida de paralelos como en la medida de meridianos (Ver Figura I.1).

Se elige:

- como paralelo 0 al círculo ecuatorial
- como meridiano 0 el círculo máximo que pasa por Greenwich

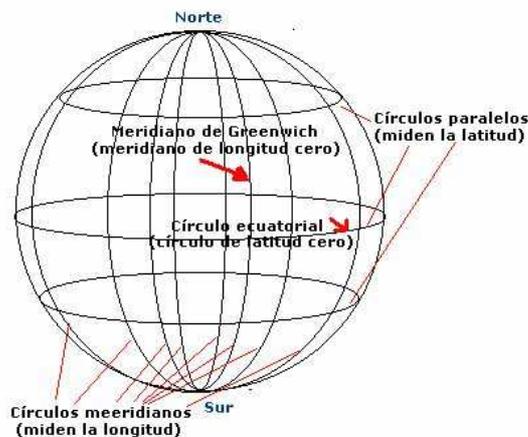


Figura I.1. Meridianos y paralelos de la esfera planetaria, China (6).

Las coordenadas geográficas son una forma de designar un punto sobre la superficie terrestre con el siguiente formato:

- 1°39'26" W
- 78°40'21" S

Esta designación supone la creación de un sistema de referencia de tres dimensiones:

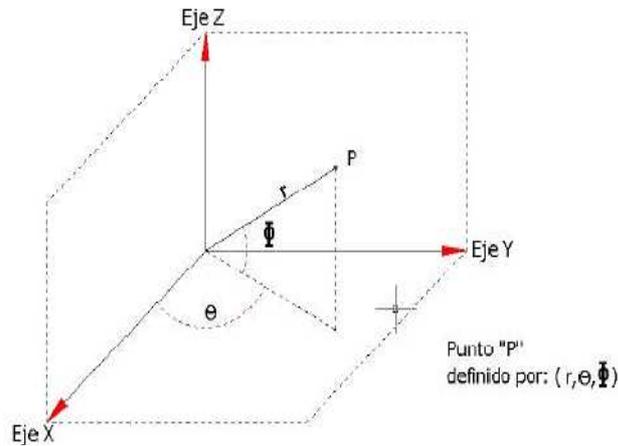


Figura 1.2. Representación del sistema tridimensional, Fernández (10)

Se define el eje de la tierra como la recta ideal de giro del globo terráqueo en su giro del movimiento de rotación. Es la recta que une los dos polos geográficos.

### 1.3.2. Localización geográfica de un punto

Básicamente la localización geográfica de un punto se puede realizar detallando uno de estos dos parámetros:

- Coordenadas geográficas en formato Longitud-Latitud.
- Coordenadas (x,y) Marcador transversal universal o UTM

Cada uno de estas dos formas de localizar un punto sobre la superficie terrestre debe de cumplir los siguientes requisitos:

- Que el punto sea único
- Que quede perfectamente identificado el sistema de proyección empleado al localizar el punto.
- Que permita referenciar la coordenada “z” del punto

Una vez que se tiene establecida una red de meridianos y paralelos, la situación geográfica de un punto viene definida por su longitud y su latitud, con referencia a la red creada.

## Longitud

Se define la Longitud ( $l$ ) de un punto  $P$  como el valor del diedro formado por el plano meridiano que pasa por  $P$  y el meridiano origen, ( $0^\circ$  Meridiano de Greenwich), Fernández (10).

La longitud es gráficamente el ángulo formado por  $OAB$ , como se puede apreciar en la figura I.3:

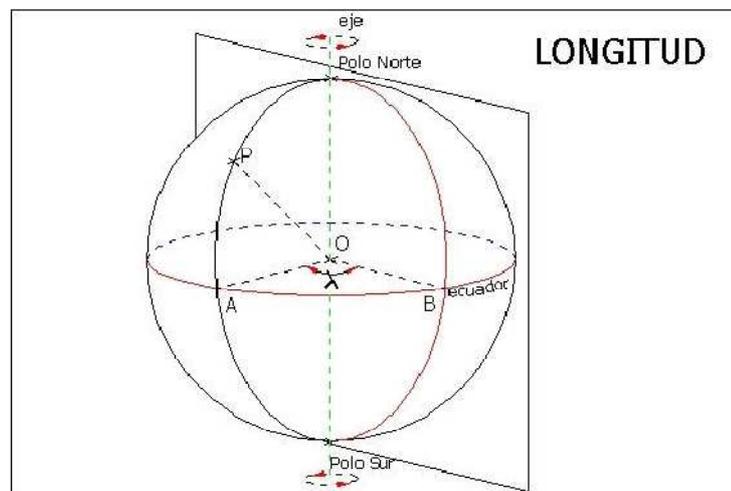


Figura I.3. Definición de longitud, Fernández (10)

La designación de la longitud lleva aparejada la designación de la posición espacial del punto con respecto al meridiano origen o meridiano de Greenwich, así se designa posición Oeste (W) cuando está a la izquierda del meridiano origen y Este (E) cuando está situado a la derecha.

La longitud presenta un mínimo posible de  $0^\circ$  hasta un máximo de  $180^\circ$ ,  $0^\circ - 180^\circ E$ ,  $0^\circ - 180^\circ W$ .

## Latitud

Se denomina Latitud geográfica ( $v$ ) de un punto  $P$  al ángulo formado por la vertical a la tierra que pasa por dicho punto con el plano ecuador, Fernández (10).

La vertical se considera la unión del punto con el origen o centro de la tierra, obteniéndose la latitud midiendo el ángulo ( $\nu$ ) sobre el meridiano que pasa por el punto P, como se puede apreciar en la figura I.4.

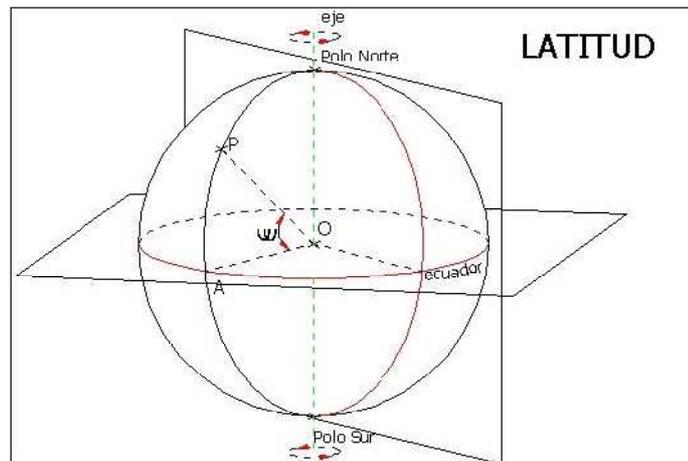


Figura I.4. Definición de latitud, Fernández (10)

La latitud máxima y mínima va desde los  $0^\circ$  hasta los  $90^\circ$ ,  $0^\circ$ - $90^\circ$ N,  $0^\circ$ - $90^\circ$ S. Los  $90^\circ$  de latitud coinciden con los polos, polo Norte y polo Sur.

## El Norte

El aparentemente simple concepto de “Norte” engloba una serie de definiciones que es necesario conocer y diferenciar adecuadamente:

- **Norte geográfico**

Es el que viene dado por la intersección del eje de rotación de la Tierra con la superficie de la misma. Es llamado también “Norte verdadero”, y en él confluyen todos los meridianos.

- **Norte magnético**

Es el punto donde la mayor parte de las líneas de fuerza del campo magnético terrestre entran en la superficie. Se puede detectar utilizando instrumentos tales como la brújula y la “flux valve”, equivalente a la brújula en las aeronaves modernas.

Es importante hacer notar que el norte geográfico y el magnético NO coinciden, y que además el norte magnético cambia su posición con el tiempo, Salazar (25).

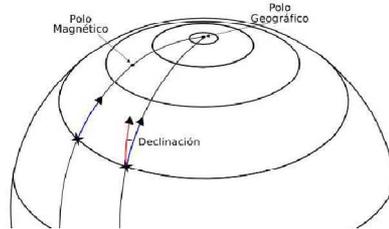


Figura I.5. Declinación de dos puntos diferentes en la Tierra, Salazar (25).

#### 1.4. El Rumbo y la Brújula

El rumbo es el ángulo horizontal que forma el meridiano que pasa por el móvil, (meridiano del lugar o local), con la trayectoria del mismo.

Según sea el meridiano de referencia, se distinguen los siguientes tipos de rumbos:

- Rv, Rumbo verdadero, toma como referencia el meridiano geográfico, y se empieza a contar a partir del norte verdadero:
- Nv. Se mide directamente en la carta náutica.
- Rm, Rumbo magnético, toma como referencia el meridiano magnético, y se mide a partir del norte magnético
- Ra, Rumbo de aguja, es el marcado por el compás magnético de abordo, que lo hace desde el norte de aguja.

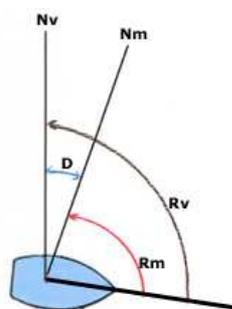


Figura I.6. La declinación magnética y el rumbo verdadero, González (11)

### 1.4.1. Sistemas de medición

Existen varias formas de medir los rumbos:

En el sistema circular, el rumbo, R, se mide de  $0^{\circ}$  a  $360^{\circ}$  en el sentido de las agujas del reloj, es decir desde el meridiano del lugar hacia el este.

En el sistema cuadrantal, el rumbo se mide desde el norte o el sur, hacia el este u el oeste. Existe por lo tanto cuatro cuadrantes, y siempre toma un valor entre  $0^{\circ}$  y  $90^{\circ}$ .

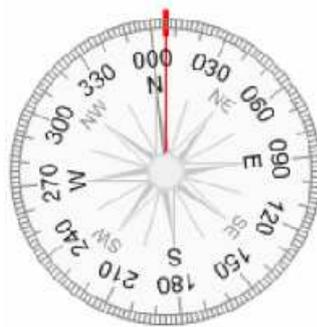


Figura 1.7. El rumbo en el sistema circular, González (11)

El sistema por cuartas, está basado en la rosa de los vientos, que divide el horizonte en 32 rumbos.

Cada uno de los cuatro cuadrantes se divide en ocho partes iguales, llamadas cuarta:

$$1 \text{ cuarta} = 11.25^{\circ} = 11^{\circ} 15'$$

### 1.4.2. La declinación magnética

Los Polos Magnéticos de la Tierra no coinciden exactamente con los Polos Geográficos situados en los extremos del imaginario eje de giro del planeta. Cabe por tanto hablar de un Norte Magnético y de un Norte Geográfico, son muy próximos pero diferenciados. Los planos se establecen en base al norte geográfico, en él convergen los meridianos, mientras que la brújula responde a la atracción de los polos magnéticos del planeta y nos señalará el norte magnético.

Esta diferencia entre norte geográfico y magnético se concreta en un pequeño ángulo al que se denomina declinación magnética, y que se representa por la letra griega  $\delta$ . La declinación magnética no es la misma en función de la posición que se ocupa sobre la Tierra y tampoco es constante en el tiempo ya que los polos magnéticos no permanecen estables sufriendo un desplazamiento lento y continuo pero siempre próximo a los polos geográficos.

La declinación magnética o variación local, es el ángulo que forma el meridiano geográfico y el magnético en un punto de La Tierra. Se cuenta desde el Norte verdadero  $N_v$  al Norte magnético  $N_m$ .

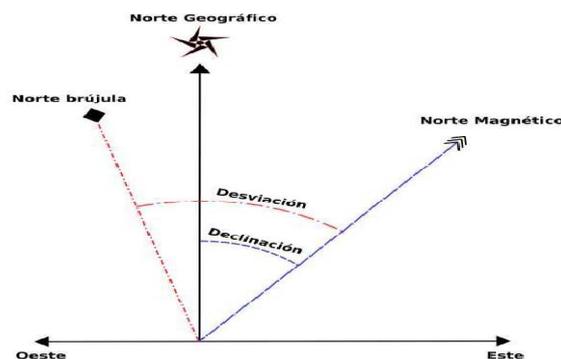


Figura 1.8. Los diferentes nortes y sus variaciones angulares Salazar (25).

## 1.5. El Sistema de Posicionamiento Global

El sistema de posicionamiento global, permite conocer la ubicación actual sobre cualquier punto de la superficie terrestre, en términos de latitud, longitud y altura.

Su nombre exacto y completo es NAVSTAR GPS (NAVigation Satellite Time

And Ranging- Global Positioning System). El proyecto data de 1973 e implica al Departamento de Defensa de los EEUU. Inicialmente, las

aplicaciones previstas del sistema eran militares; más tarde se amplió su disponibilidad a usos civiles y de investigación, Tauraso (30).

Se basa en un conjunto de 24 satélites (18 operativos y 6 en reserva) situados sobre 6 órbitas inclinadas  $55^\circ$  respecto al ecuador. Cada órbita contiene 4 satélites (3 operativos y 1 en reserva) y el entramado está pensado de forma que desde cualquier punto del globo terrestre pueda recibirse la señal de al menos seis de ellos. Las especificaciones generales del sistema prevén una cobertura de 4 a 8 satélites con elevación superior a  $15^\circ$ , lo que garantiza una buena recepción de la señal.

Están equipados con 4 relojes atómicos (dos de Cesio y dos de Rubidio) de gran precisión (el retardo de sincronización es del orden de una milmillonésima de segundo), un ordenador a bordo y un sistema de comunicaciones en muy alta frecuencia, lo que determina la precisión de las señales del sistema de satélites.

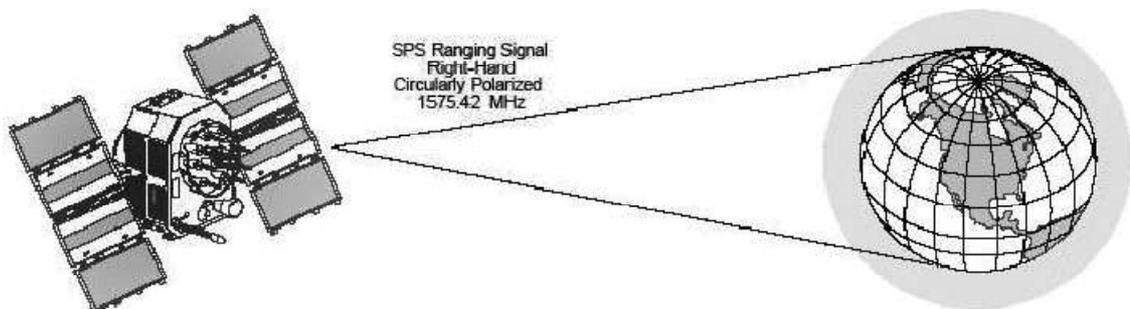


Figura I.9. Satélite GPS, Tauraso (30).

El sistema comprende tres segmentos para su funcionamiento. El segmento de control está hecho de cinco estaciones de mando localizadas alrededor del mundo. Su propósito principal es rastrear los satélites y proveerlos de corrección orbital e información del reloj. El segmento espacial lo comprenden todos los satélites y por último, el segmento del usuario simplemente es un receptor de GPS que despliega su posición en un formato conocido y fácil de interpretar.

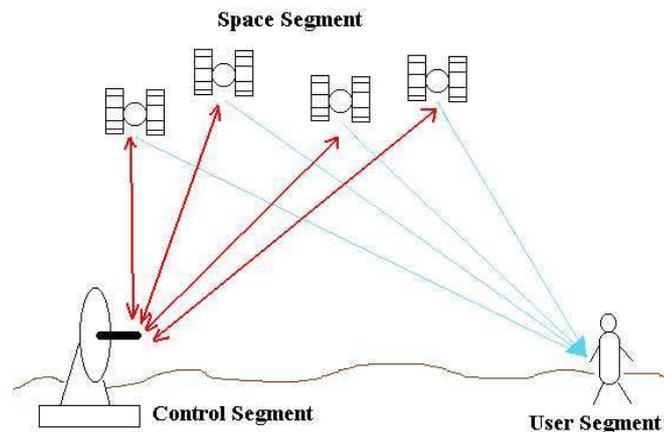


Figura I.10: Los tres segmentos del GPS, Chung Yin (8).

El control desde tierra se efectúa con un sistema llamado OCS (Operational Control System), formado por una emisora MCS (Master Control Station) y una serie de estaciones de control y antenas dispuestas a lo largo del ecuador, Chung Yin (8).

La MCS se ocupa de la sincronización de los relojes, del control de las órbitas y del diagnóstico del sistema. El procesador controla la exacta decodificación de la señal y el cálculo trigonométrico que permite determinar la posición. Los satélites transmiten en SSM (Spread Spectrum Modulation) o modulación de espectro difuso que hace que la señal resulte casi imposible de interceptar o interferir, ya que se confunde con el ruido electromagnético de fondo.

Los relojes de a bordo generan una frecuencia fundamental de 10,23 MHz a partir de la cual se producen dos portadoras básicas: L1, a 1.575,42 MHz (multiplicando la frecuencia base por 154) y L2, a 1.227,60 MHz (multiplicando por 120). La más interesante es L1, ya que se utiliza en todos los receptores comerciales.

Curiosamente, el cálculo de la posición se basa en el tiempo que tarda la señal en recorrer la distancia que separa el satélite del receptor. Para su medida, el satélite envía dos códigos PRN (Pseudo-Random Noise) con

una frecuencia de reloj de 1.023 MHz, que se repiten cada milésima de segundo.

El primero se denomina CA/Code (Coarse/Acquisition- Code) y va modulado exclusivamente sobre L1 y el segundo, llamado P-Code (Precision-Code), se modula simultáneamente sobre L1 y L2. El proyecto original considera dos sistemas de posicionamiento superpuestos: el SPS (Standard Positioning Service), para usos civiles, y el PPS (Precise Positioning Service), sólo para usos militares y ciertos usuarios autorizados.

El C/A-Code se aplica al SPS y el P-Code al PPS. En el caso del primero, el intervalo de tiempo entre la emisión y la recepción se mide comparando el código C/A enviado por el satélite con una copia que realiza el receptor, sincronizada con el satélite. Multiplicando la velocidad de la luz en el vacío (300.000 km.s) y aplicando algunas correcciones (dispersión y difracción estratosférica, efecto Doppler-Fizeau para el desplazamiento relativo satélite-receptor, etc.) se obtiene la distancia exacta entre los puntos de emisión y recepción.

El receptor GPS posee en su memoria el almanaque y la efeméride de cada satélite y sabe a cual pertenece la información que recibe gracias al ruido pseudo aleatorio que es enviado junto con los datos. Una vez recibida la información, el receptor GPS calcula la distancia existente entre el satélite y el receptor; por tanto, el receptor sabe que se encuentra dentro de una esfera cuyo centro es el satélite. Lo mismo sucederá con los demás satélites que logre contactar el receptor teniendo así esferas de posición, las cuales son intersecadas para hallar así la ubicación.

Repitiendo este mismo proceso para varios satélites (mínimo tres, para longitud y latitud) se obtiene la posición del receptor con una precisión suficiente para la navegación de vehículos. El usuario se limita a leer los datos de longitud y latitud mediante un protocolo adecuado; el más difundido es el NMEA, aunque existen otros.

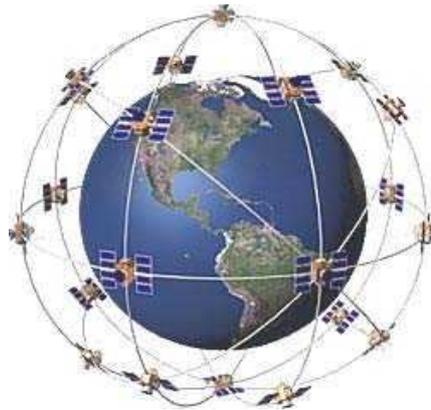


Figura I.11: La constelación de satélites de GPS que circundan la Tierra en órbitas muy precisas, Chung Yin (8).

Este servicio se impuso mundialmente y de libre uso bajo las condiciones de cualquier clima, 24 horas al día y puede aplicarse en el aire, en el mar y en la tierra. Muchos de los aviones militares y comerciales usan GPS para ayudarlos en la navegación así como también para navegar en el mar dónde el GPS se ve como una solución simple. En tierra, el GPS puede ser usado para inspección, ofreciendo tiempo de instalación reducido y precisión mejorada.

### **1.5.1. Errores en el sistema GPS**

Hay varias fuentes de error que afectan la operación del GPS:

- Retardos de la Ionosfera y la troposfera
- La atmosfera de la tierra reduce la velocidad de la señal del satélite, causando un retardo de tiempo.
- Múltiples caminos de la señal

### **1.5.2. Numero de satélites visibles.**

La precisión del GPS incrementa tanto como el número de satélites visibles al usuario crece. Esta mejora es limitada por el ambiente circundante. Las señales pueden ser bloqueadas por estructuras altas y no hay recepción dentro y bajo la tierra. Con referencia a la ubicación del usuario, el satélite

de la izquierda tiene su señal bloqueada por un edificio alto. El usuario recibe una correcta señal de un satélite cercano (a la derecha), junto con una señal reflejada adicional.

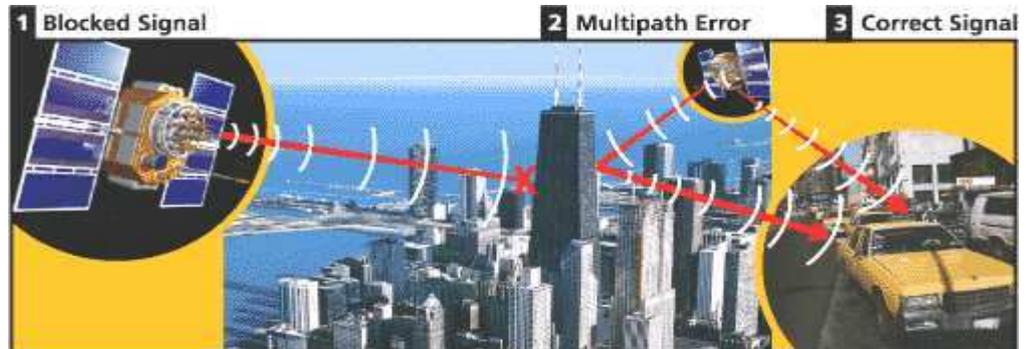


Figura I.12: Errores de GPS que muestran una señal bloqueada y una señal multi-camino, Chung Yin (8).

### 1.5.3. Desviaciones por los relojes internos de los satélites.

Esto es debido a la diferencia entre el reloj interno del receptor y el reloj atómico del satélite. Los satélites poseen relojes muy precisos, pero con el tiempo suelen presentar errores, así mismo debido al campo gravitatorio el reloj interno tiende a adelantarse. Los errores en los relojes pueden añadir hasta un metro en el rango de error.

### 1.5.4. Errores Orbitales

Estas son inexactitudes dentro de la localización de los satélites reportados, la cual es también conocida como “ephemeris errors”. Debido a la gravedad de la tierra, tormentas solares, etc. Las órbitas de los satélites se ven modificadas por lo que la distancia entre el satélite y el receptor se ve afectada, con un error que puede alcanzar hasta un metro.

## 1.6. El sistema WSG84

El sistema coordinado geodésico mundial de 1984 es un sistema terrestre convencional modificado para el sistema de navegación por satélites, con referencia a la trama Doppler NSWC 9Z-2 en origen y escala y rotada para

brindar referencia al meridiano Bureau BIH definido como meridiano zero, EUROCONTROL (9).

El origen y los ejes del sistema WGS 84 se definen como sigue:

- El origen es el centro de masa de la Tierra
- El eje Z es la dirección del polo terrestre convencional
- El eje X es la intersección del plano del meridiano BIH y el ecuador
- El eje Y es medido hacia la derecha a 90 grados del eje X

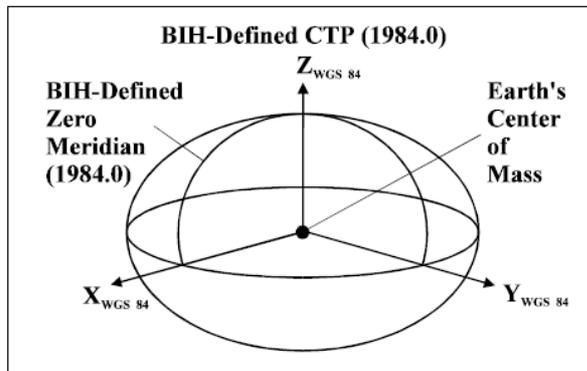


Figura I.13. Ejes del sistema WGS 84, EUROCONTROL (9).

Este sistema incluye un modelo terrestre y es definido por una serie de parámetros primarios y secundarios.

Los parámetros primarios definen un corte del elipsoide terrestre su velocidad angular y su masa la cual incluye:

Parámetros	Nombre	WGS 84
Eje semi mayor	a	6378137 m
Flattening	f	1/298.257223563
Velocidad angular	w	7.292115 x 10 <sup>-5</sup> rad s <sup>-1</sup>
Constante gravitacional	GM	398600.5 km <sup>3</sup> s <sup>-2</sup>

Tabla I.1. Parámetros principales del sistema WGS 84, EUROCONTROL (9).

Los parámetros secundarios definen en detalle el modelo de campo gravitacional de la Tierra EGM de orden y grado n=m=180.

Se puede variar el orden y el grado para obtener diferentes modelos de campo de gravedad con objetivos específicos. El WGS 84 EGM con  $n=m=41$  es el más apropiado para calcular la órbita de satélites, el más usado para satélites de navegación GPS.

El sistema utiliza estaciones de monitoreo terrestre en cinco lugares estratégicos dispuestos en todo el mundo.

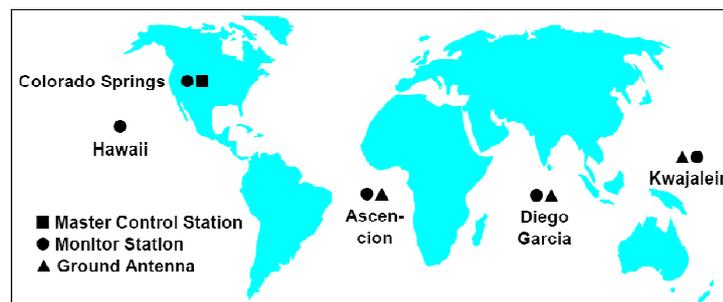


Figura I.14. Distribución de las zonas de monitoreo GPS, EUROCONTROL (9).

El sistema coordina directamente la posición de un satélite, su respectiva ephemerides de precisión y los datos de seguimiento adquiridos desde la base en términos de latitud, longitud y altura geodésica.

Estos errores colaboran en corregir su posición respecto al origen del eje de coordenadas y el centro terrestre y determinar la correcta escala. Esto no significa que se esté mejorando la precisión del GPS en modo diferencial.

### 1.7. El estándar NMEA 0183

La Asociación de Electrónica Marina Nacional (NMEA) es una asociación de sin fines de lucro de fabricantes, distribuidores, instituciones educativas, y otros interesados en la periféricos electrónicos ocupados para la marina.

El estándar NMEA 0183 define una interfaz eléctrica y protocolo para los datos de las comunicaciones entre la instrumentación marina. Es un

estándar de industria voluntaria, primero se lanzo en marzo de 1983 y ha sido actualizado hasta la versión 3.0 2, Betke (2).

NMEA también ha establecido un grupo activo para desarrollar un nuevo estándar para las comunicaciones de los datos entre los dispositivos electrónicos. La nueva norma, NMEA 2000, es una red de datos serie bidireccional, multi-transmisor, multi-receptor. Es multi-maestro y auto configurando, y no hay ningún director central NMEA (20).

### 1.7.1. Interfaz eléctrica

Los dispositivos NMEA 0183 son designados como talkers o listeners (transmisores o receptores, algunos dispositivos pueden ser los dos) empleando una interfaz serial asincrónica con los siguientes parámetros:

Velocidad Baudios	4800
# bits de datos	8
Bits de parada	1
Paridad	ninguna
Handshake	ninguna

Tabla I.2. Parámetros de comunicación serial NMEA 0183, Betke (2).

A 4800 b/s se puede enviar sólo 480 caracteres en un segundo. Dado que una frase de NMEA puede ser tan larga como 82 caracteres, limita la transmisión a menos de 6 sentencias diferentes. El límite real es determinado por las sentencias específicas usadas, pero esto muestra que es fácil de exceder las capacidades si se quiere una rápida respuesta de la sentencia.

NMEA está diseñado para ejecutarse como un proceso de fondo que envía sentencias que se capturan cuando son necesitadas por el programa usado. Algunos programas no pueden hacer esto y esos programas muestrearan la cadena de datos, entonces usan los datos para el despliegue en pantalla, y luego muestrean los datos de nuevo.

Dependiendo del tiempo necesitado para usar los datos puede haber un retraso de 4 segundos para cambiar de datos. Esto puede estar bien en algunas aplicaciones pero totalmente inaceptable en otras. Por ejemplo un automóvil que viaja a 60 mph viajará 88 pies en un segundo. Algunos segundos de retraso podrían hacer parecer al sistema entero lento y podrían causar giros incorrectos.

### **1.7.2. El Formato general de las sentencias**

Todos los datos se transmiten en forma de sentencias. Se permiten sólo caracteres de ASCII imprimibles, más CR (retorno de carro) y LF (línea de alimentación). Cada sentencia empieza con un signo "\$" y termina con <CR><LF>. Hay tres tipos básicos de sentencias: sentencias del talker, sentencias de propietario, sentencias de pregunta.

#### **Sentencias del transmisor**

El formato general para una sentencia del transmisor es:

\$tsss,d1,d2,... <CR><LF>

Las primeras dos letras que siguen al "\$" es el identificador del transmisor. Los próximos tres caracteres (sss) son el identificador de la sentencia, seguido por varios campos datos separados por comas, seguidas por un checksum optativo, y terminado por el retorno de carro/línea de alimentación. Los campos de datos se definen singularmente para cada tipo de sentencia.

Un ejemplo es:

\$HCHDM,238,M <CR><LF>

donde "HC" especifica al transmisor como un compás magnético, "HDM" especifica la cabecera del mensaje magnético. Los "238" son el valor cabecera, y "M" designa el valor de cabecera como magnético.

Una frase puede contener sobre 80 caracteres mas "\$" y CR/LF. Si los datos para un campo no están disponibles, el campo es omitido, pero las comas de delimitación si son enviadas, sin espacio entre ellas. El campo del checksum consiste de un " \* " y dos dígitos hexadecimales que representan la OR exclusiva entre todos los caracteres, pero no incluyen, el "\$" y " \* ".

### **Las sentencias propietario**

El estándar les permite a los fabricantes individuales definir los formatos de sentencias propietario. Estas frases empiezan con "\$P", después un ID del fabricante de 3 letras, seguido por cualquier dato que el fabricante desee, seguido por el formato general de sentencias estándar.

### **Las sentencias de pregunta**

Una sentencia de pregunta es el medio para que un receptor (listener) pida una sentencia en particular de un transmisor (talker). El formato general es:

\$ttllQ,sss,[CR][LF]

Los primeros dos caracteres del campo de dirección son el identificador del transmisor del solicitante y los dos caracteres próximos son el identificador del transmisor del dispositivo al que se preguntó (listener). El quinto carácter siempre es una "Q" que define el mensaje como una pregunta. El próximo campo (sss) contiene las tres letras mnemotécnicas de la sentencia que es solicitada.

El GPS transmitirá entonces esta sentencia una vez por segundo hasta que se haga otra pregunta. Estos son los Identificadores del transmisor:

Identificador	Descripción
AG	Autopilot - General
AP	Autopilot - Magnetic
CD	Communications – Digital Selective Calling (DSC)
CR	Communications – Receiver / Beacon Receiver
CS	Communications – Satellite
CT	Communications – Radio-Telephone (MF/HF)
CV	Communications – Radio-Telephone (VHF)
CX	Communications – Scanning Receiver
DF	Direction Finder
EC	Electronic Chart Display & Information System (ECDIS)
EP	Emergency Position Indicating Beacon (EPIRB)
ER	Engine Room Monitoring Systems
GP	Global Positioning System (GPS)
HC	Heading – Magnetic Compass
HE	Heading – North Seeking Gyro
HN	Heading – Non North Seeking Gyro
II	Integrated Instrumentation
IN	Integrated Navigation
LC	Loran C
P	Proprietary Code
RA	RADAR and/or ARPA
SD	Sounder, Depth
SN	Electronic Positioning System, other/general
SS	Sounder, Scanning
TI	Turn Rate Indicator
VD	Velocity Sensor, Doppler, other/general
DM	Velocity Sensor, Speed Log, Water, Magnetic
VW	Velocity Sensor, Speed Log, Water, Mechanical
WI	Weather Instruments
YX	Transducer
ZA	Timekeeper – Atomic Clock
ZC	Timekeeper – Chronometer
ZQ	Timekeeper – Quartz

Tablal.3. Identificadores de un transmisor GPS, Betke (2).

## Descifrando las sentencias de posición seleccionadas

Las sentencias de NMEA más importantes incluyen el GGA que proporciona los datos actuales, RMC que proporciona la mínima sentencia de información de GPS y el GSA que proporciona los datos de estado de Satélite.

### La variable GGA

Contiene los datos esenciales que proporcionan localización y precisión de datos 3D.

Así por ejemplo:

\$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,\*47

donde:

GGA	Datos FIX del sistema de posicionamiento global
123519	Lectura tomada a las 12:35:19 UTC
4807.038,N	Latitud 48 grados 07.038' N
01131.000,E	Longitud 11 grados 31.000' E
1	Fix quality: 0 = invalido
	1 = GPS fix (SPS)
	2 = DGPS fix
	3 = PPS fix
	4 = Tiempo real kinematico
	5 = Variable RTK flotante
	6 = Estimado (2.3 por defecto)
	7 = Modo de entrada manual

Tabla I.4. La variable GGA y su contenido, Betke (2).

## Las variables GSA - GPS DOP y los satélites activos.

Esta sentencia proporciona los detalles del fix. Incluye los números de los satélites que son usados en la solución actual y el factor de dilución de precisión DOP. El DOP es un indicador del efecto de geometría del satélite en la exactitud del fix. Es un número sin unidad donde el más pequeño es mejor. Para fix 3D que usan 4 satélites un 1.0 podría ser considerado un número perfecto, sin embargo para las soluciones sobre- determinadas es posible ver números por debajo de 1.0.

Ejemplo:

```
$GPGSA,A,3,04,05,,09,12,,,24,,,,,2.5,1.3,2.1*39
```

Donde:

GSA	Estado del satélite
A	Auto selección of 2D or 3D fix (M = manual)
3	3D fix – valores incluyen: 1 = no fix
	2 = 2D fix
	3 = 3D fix
04,05...	PRNs de satellites usados para el fix (espacio para 12)
2.5	PDOP (dilución de precisión)
1.3	Precisión de dilución horizontal (HDOP)
2.1	Precisión de dilución vertical (VDOP)
*39	datos de checksum que inicia siempre con *

Tabla I.5. La variable GSA y su contenido, Betke (2).

## La variable GSV

Los Satélites a la vista GSV muestran los datos sobre los satélites que la unidad puede ser capaz de encontrar viendo sus datos de máscara y almanaque. También muestra la habilidad de rastrear estos datos. Una sentencia de GSV sólo puede proporcionar los datos para 4 satélites y se

puede necesitar de 3 sentencias para completar la información. Es razonable para la sentencia GSV contener más satélites que GGA podrían indicar, ya que GSV puede incluir satélites que no se usan como la parte de la solución. No es un requerimiento que todas las sentencias GSV aparezcan en secuencia.

En el estándar NMEA está a menudo relacionada con la fuerza de la señal. SNR es un valor indirecto pero más útil que la fuerza de la señal sin tratar. Puede ir de 0 a 99 y tiene unidades de dB según el estándar NMEA. El rango de valores de trabajo en un GPS dado normalmente mostrará una diferencia de aproximadamente 25 a 35 entre los valores más bajos y más altos, sin embargo 0 es un caso especial y puede estar mostrando satélites que están a la vista pero están rastreándose.

Ejemplo:

\$GPGSV,2,1,08,01,40,083,46,02,17,308,41,12,07,344,39,14,22,228,45\*75

Donde:

GSV	Satélites a la vista
2	Número de sentencias para completar los datos
1	Sentencia 1 de 2
08	Numero de satélites a la vista
01	Numero del satélite PRN
40	Grados de elevación
83	Grados de Azimuth
46	SNR entre más alto mejor para un número de satélites mayor a 4
*75	Los datos de checksum

Tabla I.6. La variable GSV y su contenido, Betke (2).

## Los valores de RMC - NMEA

RMC tiene su propia versión de datos para pvt (posición, velocidad, tiempo) esenciales del GPS. Se llama RMC, El Mínimo Recomendado es similar a:

GPRMC,123519,A,4807.038,N,01131.000,E,022.4,084.4,230394,003.1,W\*6<sup>a</sup>

Donde:

RMC	Sentencia del mínimo recomendado
123519	Lectura tomada a las 12:35:19 UTC
A	Estado A=activo o V=no determinado
4807.038,N	Latitud 48 grados 07.038' N
01131.000,E	Longitud 11 grados 31.000' E
022.4	Velocidad sobre tierra en nudos
084.4	Angulo de rumbo en grados
230394	Fecha actual 23 marzo 1994
003.1,W	Variación magnética
*6A	Datos de checksum

Tabla I.7. Los valores de RMC y contenido, Betke (2).

## La variable GLL

La Latitud y Longitud geográfica son datos importantes de la posición del satélite que luego de procesarlos darán por resultado la posición del receptor en tierra. Ejemplo:

\$GPGLL,4916.45,N,12311.12,W,225444,A,\*31

Donde:

GLL	Posición geográfica en latitud y longitud
4916.46,N	Latitud 49 grados. 16.45 min. Norte
12311.12,W	Longitud 123 grados. 11.12 min. Oeste
225444	Lectura tomada a las 22:54:44 UTC
A	Datos activos (A) o no determinados (V)
*31	Datos de checksum

Tabla I.8. La variable GLL y su contenido, Betke (2).

## La variable VTG

La variable de la mejor ruta. El receptor del GPS puede usar el prefijo LC en lugar de GP si está emulando el rendimiento de Loran.

\$GPVTG,054.7,T,034.4,M,005.5,N,010.2,K\*33

Donde:

VTG	Mejor ruta y velocidad en tierra
054.7,T	Mejor ruta en grados
034.4,M	mejor ruta magnética
005.5,N	velocidad en tierra en nudos
010.2,K	velocidad en tierra en kilómetros por hora
*33	Datos del Checksum

Tabla I.9. La variable VTG y su contenido, Betke (2).

## 1.8. Ultrasonido

Los ultrasonidos son antes que nada sonido, exactamente igual que el que se escucha normalmente, salvo que tienen una frecuencia mayor que la máxima audible por el oído humano. Aquellos sonidos que superan los 40 Khz son a los que se conoce como ultrasonidos.

El funcionamiento básico de los ultrasonidos como medidores de distancia se muestra de una manera muy clara en el siguiente esquema (Figura

l.15), donde se tiene un receptor que emite un pulso de ultrasonido que rebota sobre un determinado objeto y la reflexión de ese pulso es detectada por un receptor de ultrasonidos:

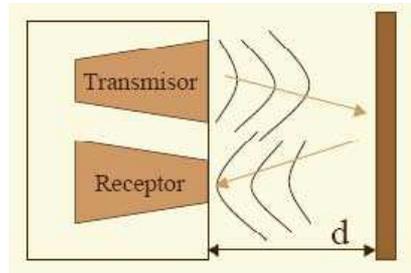


Figura 1.15. Reconocimiento por emisión recepción, Pérez (23).

La mayoría de los sensores de ultrasonido de bajo coste se basan en la emisión de un pulso de ultrasonido cuyo lóbulo, o campo de acción, es de forma cónica. Midiendo el tiempo que transcurre entre la emisión del sonido y la percepción del eco se puede establecer la distancia a la que se encuentra el obstáculo que ha producido la reflexión de la onda sonora, mediante la fórmula:

$$d = \frac{1}{2} V \cdot t \quad (1.1)$$

Donde  $V$  es la velocidad del sonido en el aire y  $t$  es el tiempo transcurrido entre la emisión y recepción del pulso.

### 1.8.1. Características

El sonido ultrasónico es una vibración con una frecuencia sobre el rango del oído humano, normalmente mayor a 20 kHz. Los micrófonos y altavoces que reciben y transmiten el sonido ultrasónico son llamados transductores. La mayoría de los sensores ultrasónicos usan un solo transductor para transmitir el pulso de sonido y recibir el eco reflejado, operan típicamente a frecuencias entre 40 kHz y 250 kHz. Una gran variedad de diferentes tipos de transductores se usan en estos sistemas. Las secciones siguientes proporcionan una apreciación global de cómo el

pulso de sonido es afectado por algunas de las propiedades ultrasónicas fundamentales del medio en los que viaja el sonido.

### **El ruido del fondo**

El nivel de ruido ultrasónico de fondo disminuye tanto como la frecuencia crece. La razón es que menos ruido es producido en el ambiente a las frecuencias más altas, y el ruido que se produce se atenúa a mayor rapidez mientras viaja a través del aire, Massa (16).

### **Los efectos de la frecuencia, distancia y transmisión**

En un sensor ultrasónico, el transductor produce un pulso corto de sonido. La magnitud de la presión de sonido generada variará de un tipo de sensor a otro. En la acústica, se expresa la presión de sonido en decibelios debido a sus grandes rangos dinámicos.

### **Niveles de eco de una superficie llana**

Si el pulso es reflejado de una superficie llana grande, entonces todo el haz se refleja. Esta reflexión total es equivalente a una fuente virtual en dos veces la distancia. Por consiguiente, la pérdida de propagación para el sonido reflejado de una superficie llana grande es igual a  $20 \log (2R)$ , y la pérdida de absorción es igual a  $2\alpha R$  donde  $R$  es la distancia del sensor al objeto y  $\alpha$  es el factor de absorción del material.

## **1.8.2. Fuentes de errores en el manejo de ultrasonidos**

A pesar de que su funcionamiento parece muy sencillo, existen factores inherentes tanto a los ultrasonidos como al mundo real, que influyen de una

forma determinante en las medidas realizadas. Por tanto, es necesario un conocimiento de las diversas fuentes de incertidumbre que afectan a las medidas para poder tratarlas de forma adecuada, minimizando su efecto en el conocimiento del entorno que se desea adquirir. Entre los diversos factores que alteran las lecturas que se realizan con los sensores de ultrasonido cabe destacar:

- El campo de actuación del pulso que se emite desde un transductor de ultrasonido tiene forma cónica. El eco que se recibe como respuesta a la reflexión del sonido indica la presencia del objeto más cercano que se encuentra dentro del cono acústico y no especifica en ningún momento la localización angular del mismo. Aunque la máxima probabilidad es que el objeto detectado esté sobre el eje central del cono acústico, la probabilidad de que el eco se haya producido por un objeto presente en la periferia del eje central no es en absoluto despreciable y ha de ser tenida en cuenta y tratada convenientemente.

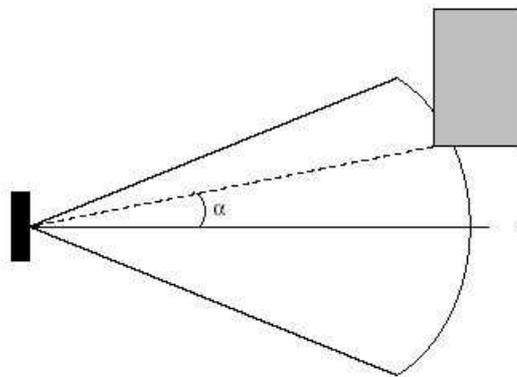


Figura I.16. Incertidumbre angular en la medida de un ultrasonido, Pérez (23).

- La cantidad de energía acústica reflejada por el obstáculo depende en gran medida de la estructura de su superficie. Para obtener una reflexión altamente difusa del obstáculo, el tamaño de las irregularidades sobre la superficie reflectora debe ser comparable a la longitud de onda de la onda de ultrasonido incidente.

- En los sensores de ultrasonido de bajo coste se utiliza el mismo transductor como emisor y receptor. Tras la emisión del ultrasonido se espera un determinado tiempo a que las vibraciones en el sensor desaparezcan y esté preparado para recibir el eco producido por el obstáculo. Esto implica que existe una distancia mínima  $d$  (proporcional al tiempo de relajación del transductor) a partir de la cual el sensor mide con precisión. Por lo general, todos los objetos que se encuentren por debajo de esta distancia,  $d$ , serán interpretados por el sistema como que están a una distancia igual a la distancia mínima.

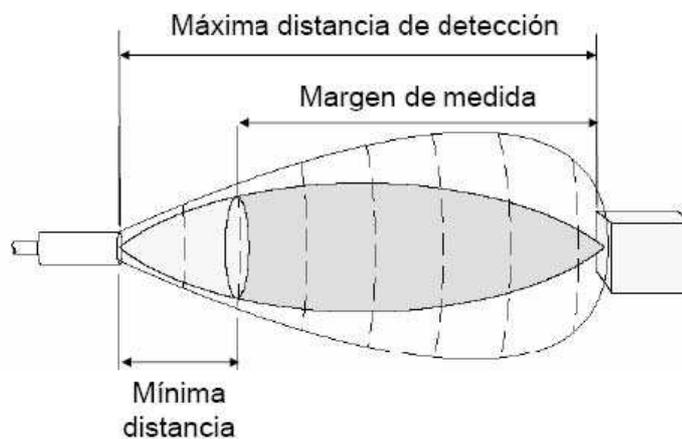


Figura I.17. Distancia mínima de reconocimiento, Pérez (23).

- Un factor de error muy común es el conocido como falsos ecos. Estos falsos ecos se pueden producir por razones diferentes: Puede darse el caso en que la onda emitida por el transductor se refleje varias veces en diversas superficies antes de que vuelva a incidir en el transductor (si es que incide). Este fenómeno, conocido como reflexiones múltiples, implica que la lectura del sensor evidencia la presencia de un obstáculo a una distancia proporcional al tiempo transcurrido en el viaje de la onda; es decir, una distancia mucho mayor que a la que está en realidad el obstáculo más cercano, que pudo producir la primera reflexión de la onda.
- Otra fuente más común de falsos ecos, conocida como crosstalk, se produce cuando se emplea un cinturón de ultrasonidos donde una

serie de sensores están trabajando al mismo tiempo. En este caso puede ocurrir (y ocurre con una frecuencia relativamente alta) que un sensor emita un pulso y sea recibido por otro sensor que estuviese esperando el eco del pulso que él había enviado con anterioridad (o viceversa).

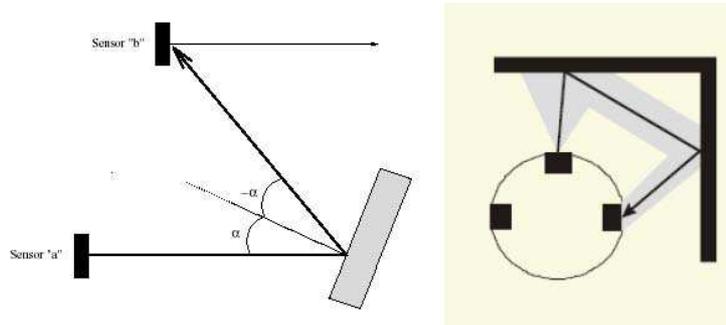


Figura 1.18. El sensor "a" emite el pulso que recibe el sensor "b", Pérez (23).

- Las ondas de ultrasonido obedecen a las leyes de reflexión de las ondas, por lo que una onda de ultrasonido tiene el mismo ángulo de incidencia y reflexión respecto a la normal a la superficie. Esto implica que si la orientación relativa de la superficie reflectora con respecto al eje del sensor de ultrasonido es mayor que un cierto umbral, el sensor nunca recibe el pulso de sonido que emitió.

Además de los problemas ya señalados, a continuación se muestran otras situaciones que pueden ser problemáticas a la hora de diseñar un sistema de detección de distancias para un robot basado en ultrasonidos:

- La posición real del objeto es desconocida: (cualquier posición del cono a distancia  $d$ ).
- Reflejos especulares: la dirección del reflejo depende del ángulo de Incidencia
- Cuanto menor sea el ángulo, mayor es la probabilidad de perderse y producir falsas medidas de gran longitud
- Las superficies pulidas agravan el problema (las rugosas producen reflejos que llegan antes)

### 1.8.3. Sensores de ultrasonido

Sensores ultrasónicos son comúnmente usados para una amplia variedad de aplicaciones de medición de presencia, proximidad, o distancia sin contacto. Estos dispositivos típicamente transmiten una corta ráfaga de sonido ultrasónico hacia un objeto, el cual refleja el sonido de regreso al sensor. El sistema luego mide el tiempo del eco que para retornar al sensor y calcula la distancia al objeto usando la velocidad del sonido en el medio.

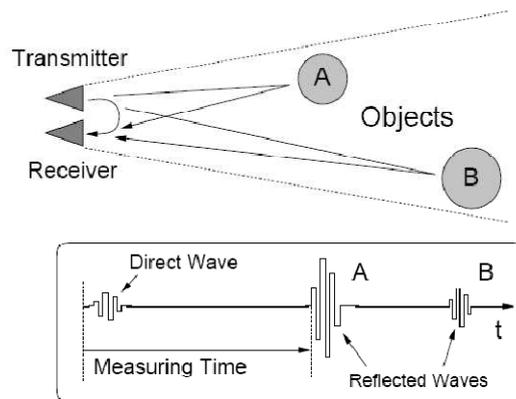


Figura I.19. Funcionamiento básico del sensor ultrasónico, Ohya (22).

Una amplia gama de sensores difieren por el tipo de configuración y características electrónicas. Acústicamente, operan a diferentes frecuencias y tienen diferentes lóbulos de radiación.

## **CAPITULO II**

### **DISEÑO MECÁNICO**

En este capítulo se presentan las características físicas, geométricas y mecánicas del robot móvil. También se detallan las diferentes situaciones que llevaron a la selección de los materiales para la construcción del sistema de tracción para el robot del presente trabajo.

#### **2.1. Modos de locomoción**

Existe una gran variedad de modos de moverse sobre una superficie sólida, entre los robots, las más comunes son las ruedas, las cadenas y las patas.

Los vehículos de ruedas son, con mucho, los más populares por varias razones prácticas. Los robots con ruedas son más sencillos y más fáciles de construir, la carga que pueden transportar es mayor, relativamente.

Tanto los robots basados en cadenas como en patas se pueden considerar más complicados y pesados, que los robots de ruedas para una misma

carga útil. A esto se puede añadir el que se pueden transformar vehículos de ruedas de radio control para usarlos como bases de robots, Sandin (26).

### 2.1.1. Ruedas de direccionamiento

Se utilizan con frecuencia ruedas de direccionamiento que no poseen control alguno y su función principal es la de sostener la estructura y brindar libertad de movimiento cuando se realizan giros, aunque se puede adaptar fácilmente un sistema que les otorgue direccionamiento controlado.

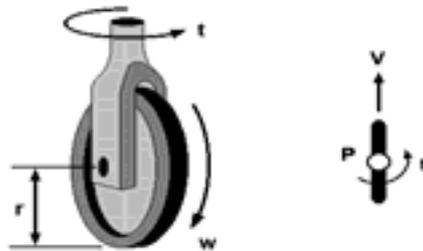


Figura II.1. Rueda de direccionamiento, Sandin (26).

### 2.1.2. Modelo de triciclo

El diseño en triciclo tiene unas prestaciones similares a un diseño de coche de cuatro ruedas con la ventaja de ser mecánicamente más simple ya que el coche necesita alguna unión entre las ruedas direccionables. En general en estos dos diseños las ruedas direccionables no son motrices, y no es necesario controlar la velocidad de las ruedas para que el robot se mantenga recto. La forma de un robot puede tener un gran impacto en sus prestaciones, un robot no cilíndrico corre mayor riesgo de quedar atrapado por una disposición desfavorable de obstáculos o de fallar en encontrar un camino en un espacio estrecho o intrincado.

Se considera dos robots del mismo tamaño uno cilíndrico y el otro cuadrado, ambos encuentran un paso estrecho según se mueven. Un algoritmo sencillo permitirá al robot cilíndrico pasar, el robot choca, gira y lo intenta de nuevo hasta que pasa. Esto es así, porque el robot es capaz de

girar estando en contacto con el obstáculo. El robot cuadrado, por el contrario, tiene que retroceder y girar y repetirlo, siempre se requiere un algoritmo más complejo para la navegación de un robot cuadrado que para la de uno cilíndrico.

### 2.1.3. Locomoción de robots móviles

Se tiene a disposición diferentes tipos de configuraciones de las ruedas para lograr ciertas características en la locomoción del móvil, donde el punto de intersección de los ejes de las ruedas determina esta característica teniendo dos configuraciones principales, el centro instantáneo de rotación y el centro instantáneo de curvatura.

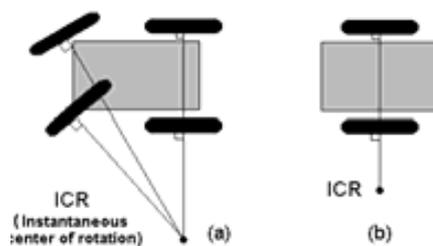


Figura II.2. a) centro instantáneo de rotación b) centro instantáneo de rotación y curvatura , Sandin (26).

### Grados de movilidad

Determina los grados de libertad de movimiento que posee el robot, los determina la posición de los ejes de las ruedas que en la tracción del móvil.

**Sin grados de libertad.-** Cuando el centro instantáneo de rotación no se puede encontrar, es decir el robot móvil es incapaz de moverse



Figura II.3. Robot con 0 grados de libertad, Sandin (26).

**Grado de libertad 1.-** cuando el móvil posee un ángulo fijo de dirección

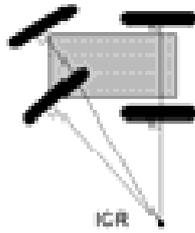


Figura II.4. Robot con 1 grados de libertad, Sandin (26).

**Grado de libertad 2.-** Cuando el móvil presenta dos centros instantáneos de rotación, es decir puede adquirir dirección variable

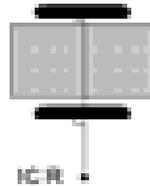


Figura II.5. Robot con 2 grados de libertad, Sandin (26).

**Grado de libertad 3.-** Cuando el móvil es capaz de encontrar su centro instantáneo de rotación siempre y localizarlo en cualquier posición.

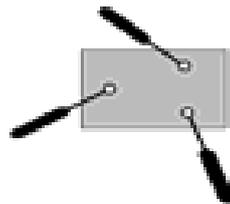


Figura II.6. Robot con 3 grados de libertad, Sandin (26).

### **Grados de direccionamiento**

El es número de ruedas centradas orientables e independientes que son capaces de dar la dirección al móvil

**Grado de direccionamiento 0.-** Cuando el robot móvil posee una rueda centrada sin capacidad mecánica de rotación.

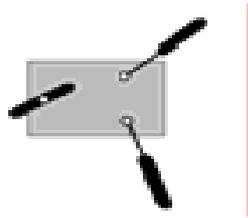


Figura II.7. Robot con 0 grados de direccionamiento, Sandin (26).

**Grado de direccionamiento 1.-** Cuando el móvil posee dos ruedas mutuamente dependientes y que son controlables

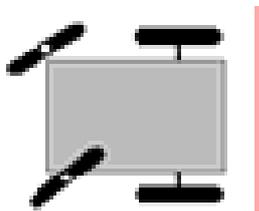


Figura II.8. Robot con 1 grado de direccionamiento, Sandin (26).

**Grado de direccionamiento 2.-** Sucede cuando el móvil posee dos ruedas de direccionamiento completamente controlables e independientes

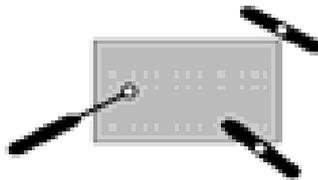


Figura II.9. Robot con 2 grados de direccionamiento, Sandin (26).

### Grados de maniobrabilidad

Es la suma de los grados de libertad mas los grados de direccionamiento que posee el robot móvil y determinan la capacidad que tiene de moverse por el entorno al que se le asigne

$$\delta_M = \delta_m + \delta_s \quad (2.1)$$

## **2.2. Plataforma de locomoción**

El robot del presente proyecto requiere de una plataforma para albergar en su interior el sistema electrónico de control, el sistema de alimentación y los motores para su locomoción, se precisa entonces del estudio de la mecánica del movimiento que se necesita para que cumpla con los objetivos, así como también de la cinemática involucrada en mover el peso que se acumula entre los diferentes sistemas descritos, de las necesidades de precisión en cuanto a velocidad y rotación y de la autonomía requerida para que el robot móvil pueda terminar con éxito la tarea encomendada.

## **2.3. Sistema de tracción**

Un sistema de tracción es un mecanismo formado por varios pares de engranajes acoplados de tal forma que el elemento conducido de uno de ellos es el conductor del siguiente. Suele denominarse como la cadena cinemática formada por varias ruedas que ruedan sin deslizar entre sí; o bien como cualquier sistema de ejes y ruedas dentadas que incluya más de dos ruedas o tandem de ejes y ruedas dentadas.

Se recurre a ellos porque no es posible establecer una determinada relación de transmisión entre dos ejes mediante un solo par de ruedas dentadas; o también porque se desea obtener un mecanismo con relación de transmisión variable, lo que tampoco es posible con un solo par de ruedas.

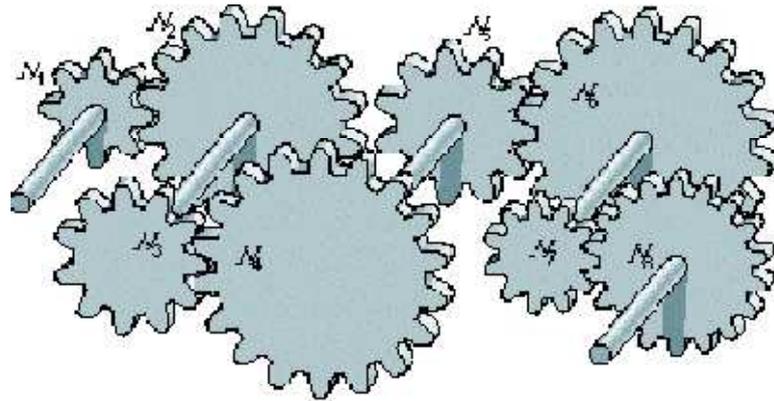


Figura II.10. Sistema de tracción mediante engranajes

El sistema de tracción está constituido por un motor DC (Corriente Directa) de 12V y un juego de engranes que se encargan de transformar las revoluciones del motor en fuerza de tracción, lastimosamente no se conocen sus características técnicas por lo que no se puede saber la relación de cambio. Estos motores rotan  $360^\circ$  a alta velocidad pero con poco torque, en este sentido fue necesario utilizar un sistema de engranajes que permitiera el desplazamiento del robot a una velocidad moderada.

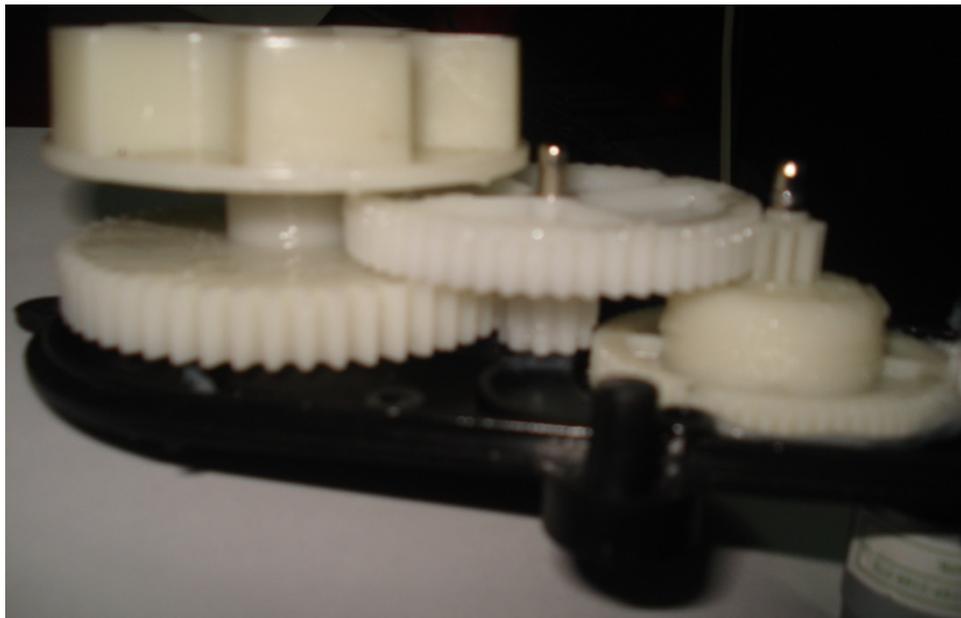


Figura II.11. Tren de engranajes usados en el robot

Se decidió utilizar la configuración de las ruedas conocida como "Differential Drive" o configuración diferencial, que consiste en colocar dos ruedas en el mismo eje con motores independientes; de esta manera cuando se hace girar los dos motores en un sentido, el robot se desplaza en dicho sentido, por el contrario si un motor se hace girar en un sentido y el otro en el sentido opuesto, entonces el robot dará vueltas sobre su propio eje de rotación. Por último si un motor se hace girar en un sentido y el otro también pero con diferencia en el número de revoluciones de cada motor, entonces el robot se desplazará describiendo una trayectoria circular.

Para lograr una velocidad suficiente con el torque necesario para mover todo el sistema se opto por conseguir unos sistemas de tracción prediseñados, usados en juguetes eléctricos, se escogió un auto infantil que puede llevar a cuestas un niño de 30 Kg sin problemas con baterías estándar de plomo de 12V 10A por 1 hora, estos sistemas constan de un motor DC mas 6 engranajes empotrados en una caja de plástico duro.

Según la mecánica de engranajes se dice que la relación entre el engrane (a) con un número de dientes  $n_1$  y el engrane (b) con un número de dientes  $n_2$ , es:

$$n_1 \cdot z_1 = n_2 \cdot z_2 \quad (2.2)$$

Donde  $n$  es el numero de vueltas y  $z$  el número de dientes de cada engranaje.

Tomando una referencia para facilitar el cálculo asumiendo una vuelta del segundo engranaje, se pudo calcular el número de vueltas del primero conociendo el número de dientes de ambos:

$$n_1 = \frac{z_2}{z_1} \quad (2.3)$$

Si un engrane tiene 8 dientes (a) y el segundo posee 24 (b), el primero entonces gira 3 veces cuando el segundo engranaje da una vuelta.

Para un sistema múltiple o tren de engranajes se emplea esta fórmula con las siguientes modificaciones:

$$n_1 = \frac{z_{salida1} \cdot z_{salida2} \cdot z_{salida\ n}}{z_{entrada1} \cdot z_{entrada2} \cdot z_{entrada\ n}} \quad (2.4)$$

Siendo  $z_{salida}$ , el número de dientes del engrane conducido y  $z_{entrada}$  el número de dientes del engrane conductor.

El sistema para el robot posee en el primer par de engranes  $z_1=12$  y  $z_2=68$ , dado que este segundo engranaje es compuesto, es decir tiene en su eje otro engranaje, a este se le toma como otro independiente, entonces, el segundo par transmite la fuerza con engranes de 8 y 54 y el tercero ofrece un cambio con engranajes de 10 y 51, la fórmula da como resultado 195,075 vueltas para el engranaje que se encuentra acoplado al motor, que quiere decir que por cada 195 vueltas aproximadamente del motor, la rueda del robot gira una, lo que en términos de torque significa una relación velocidad fuerza adecuada para mover toda la estructura sin inconvenientes.



Figura II.12. Sistema de tracción

Como ya se discutió con anterioridad, esta característica se concluyó luego de realizar pruebas de campo, debido a que no se conocen las características técnicas de los motores usados y solamente se asumió sus capacidades aparentes cuando se encuentran funcionando en los vehículos infantiles para los cuales se les fabrica.

#### **2.4. Diseño de la estructura**

Se decidió trabajar la plataforma donde se montará el sistema mecánico de tracción y el sistema de control electrónico sobre tubo cuadrado de aluminio basándonos en sus características de peso, resistencia y propiedades magnéticas pues es fundamental mantener aislado el modulo de compas electrónico de cualquier fuente de campos magnéticos parásitos que puedan interferir con la correcta lectura de los campos terrestres y por consecuencia la precisión del cálculo del rumbo.



Figura II.13. Material de aluminio de 2 pulgadas para la estructura

El hierro como material común para construir estructuras posee características de resistencia y facilidad de manejo, aparte de ser una opción menos costosa, pero es muy fácil de magnetizar y en este caso mucho mas, pues la estructura está en contacto muy cercano con los imanes de los motores DC que se utilizan en la etapa de tracción, según la física básica, el hierro tiende a adquirir las características magnéticas de los imanes con suma facilidad y permanecer en este estado dependiendo de sus componentes constituyentes. Siendo este el caso el problema radica en que los campos generados por la estructura son capaces de interferir con las lecturas de los campos magnéticos terrestres a cargo de la

brújula electrónica e impedir que se pueda determinar la navegación del móvil en la superficie terrestre.

Los motores por otra parte, cuando entran en funcionamiento, el campo magnético, tanto de los imanes permanentes como de las bobinas del rotor se encierran en la estructura del mismo lo que hace que prácticamente desaparezcan y no puedan molestar en las lecturas.



Figura II.14. Construcción del chasis del robot

Por estas razones se eligió el aluminio, por su poco peso que colabora en el ahorro de energía, por sus características anti magnetizables y por su fortaleza, la estructura resultante es capaz de albergar el sistema mecánico de tracción y el sistema de control electrónico.

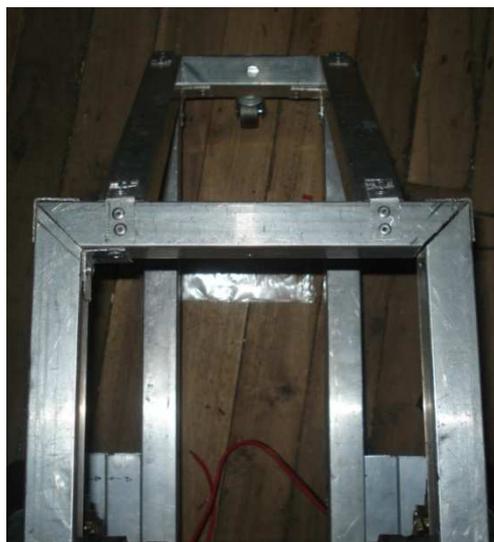


Figura II.15. Estructura de aluminio del robot

## **CAPITULO III**

### **DISEÑO ELECTRÓNICO**

En este capítulo se detallan las características electrónicas de los dispositivos elegidos para la construcción del robot. También se explican los conceptos tomados en cuenta para el diseño de cada una de las partes que conforman el proyecto.

#### **3.1. Esquema general del sistema**

El sistema comprende la conexión mediante buses de datos, de módulos creados separadamente con el fin de individualizar las posibles fuentes de falla, ya que al construirlas y probarlas una por una, se tendrá la certeza que al momento de unir las todas, el sistema funcionara como se desea, además de respetar cada uno de los protocolos de comunicación serial que se utilizan para transmitir datos entre dichos módulos.

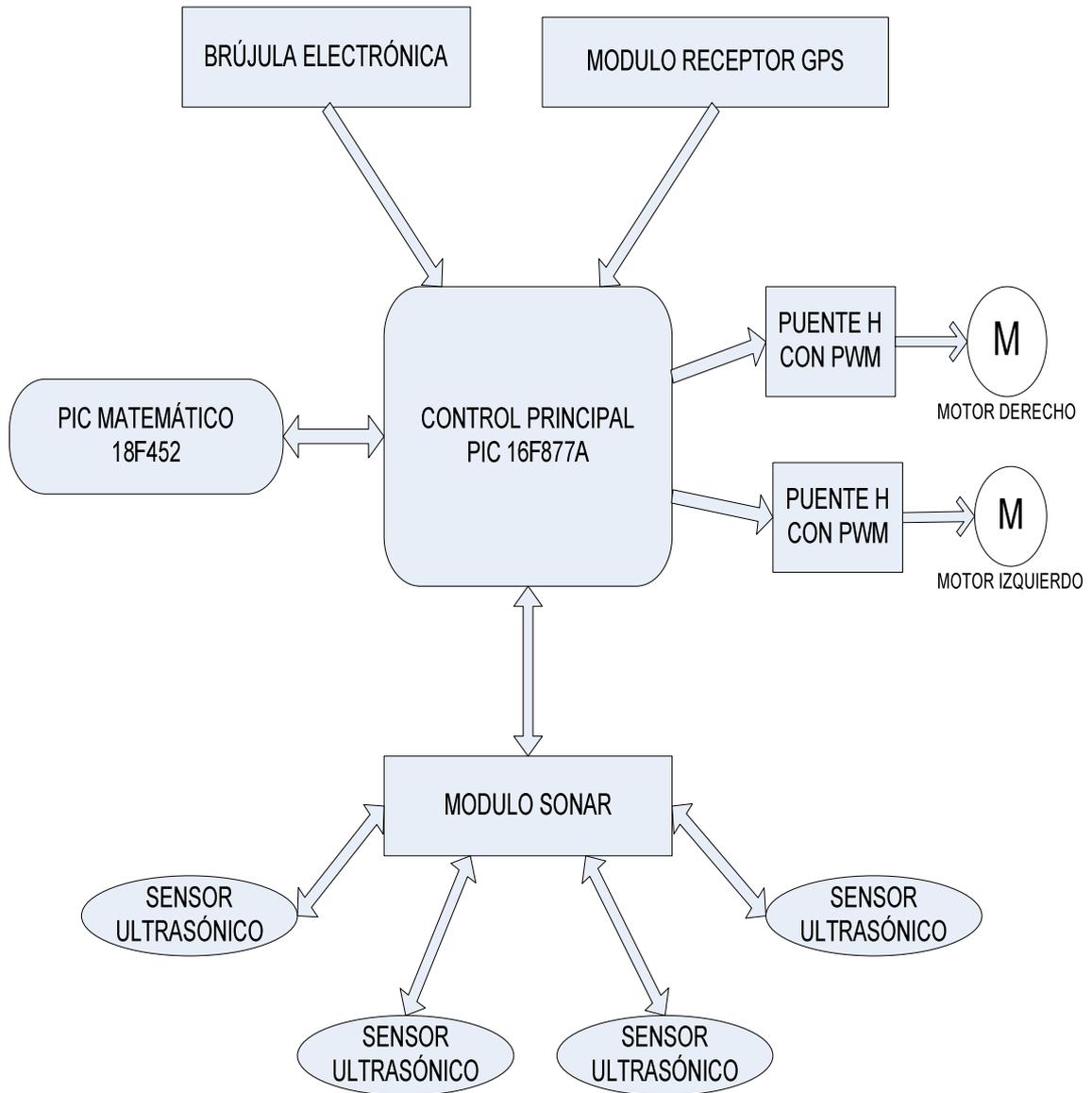


Figura III.1. Diagrama de bloques del sistema robótico completo.

### 3.2. Fuente de alimentación

Una parte esencial del robot es el sistema electrónico, que necesita una fuente de energía y dado que un factor importante del diseño es la autonomía, se le dotó de toda la corriente necesaria a través una batería del tipo acumulador de plomo.



Figura III.2. Batería de 12 voltios a 7 amperios

La batería debe suministrar toda la potencia necesaria para cubrir las necesidades de los dos motores de DC, sensores ultrasónicos, modulo GPS, brújula electrónica, display LCD, micro-controladores PIC entre otros dispositivos electrónicos. El sistema de alimentación consta de una batería que proporciona 12V a 7Ah y la electrónica que permite obtener las tensiones necesarias para cada parte del sistema robótico

### 3.2.1. Módulo regulador y distribuidor de voltaje

Es la tarjeta encargada de suministrar todos los voltajes necesarios para el correcto funcionamiento del robot. Dispone de una entrada para los 12V que ofrece la batería. Como salidas, se obtienen las tres diferentes tensiones que requiere el robot:



Figura III.3. Placa de alimentación de múltiples salidas

- Salidas de 12V.- Tensión de salida con característica de alta corriente que se aplica a los motores de DC de tracción.
- Salidas de 5V.- Voltaje que se utiliza para los sensores ultrasónicos, brújula y toda la electrónica digital en general. Todos los circuitos integrados que se emplean en el robot funcionan con valores digitales estándar de 5V para el uno lógico y 0V para el cero lógico. El voltaje se obtiene usando un regulador integrado de la serie 7805, que nos ofrece un voltaje estabilizado y regulado de 5V a 1A máximo, siempre y cuando posea un disipador refrigerante adecuado.
- Salida de 3.2V.- Voltaje necesario para activar el módulo GPS. Se emplea el regulador de tensión variable LM317 que es ajustado mediante un potenciómetro multi-vuelta del tipo trimmer hasta llegar al valor deseado, con corriente máximo de 1.5A, tras esta etapa se colocan capacitores electrolíticos los que filtran los posibles ruidos provenientes de las otras etapas, permitiendo así un aislamiento adecuado entre módulos y evitando errores o activaciones incorrectas.

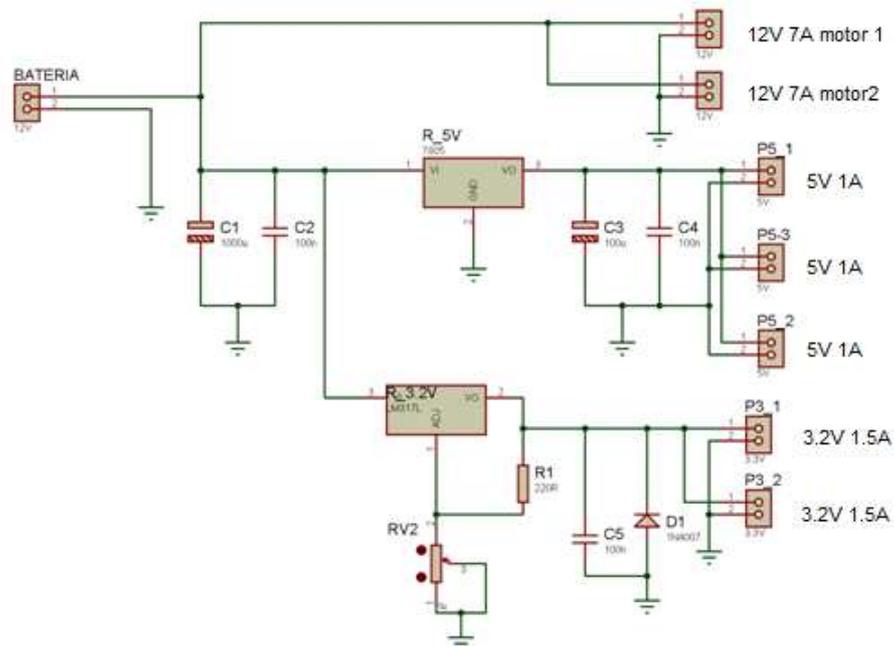


Figura III.4: Diagrama del circuito electrónico para la tarjeta de alimentación

### 3.3. Circuito interruptor digital

Es un circuito electrónico auxiliar que permite conmutar cargas de alta corriente con un simple pulso digital, es el encargado de encender y apagar todo el robot.



Figura III.5. Modulo del interruptor digital

Su funcionamiento está basado en dos flip-flop tipo D, el primer flip-flop tiene conectada su entrada "S" (star) a un transistor y un capacitor de 4.7uF, el transistor sirve como interruptor electrónico que hace que el capacitor se descargue y cargue, generando un tiempo para que el primer flip-flop tenga un pulso a su salida "Q", este se conecta al segundo flip-flop que activa a través de su salida "Q" un transistor que hace que el relé conmute y permite que el robot sea encendido o apagado, es decir el robot se activa al presionar un pulsador.

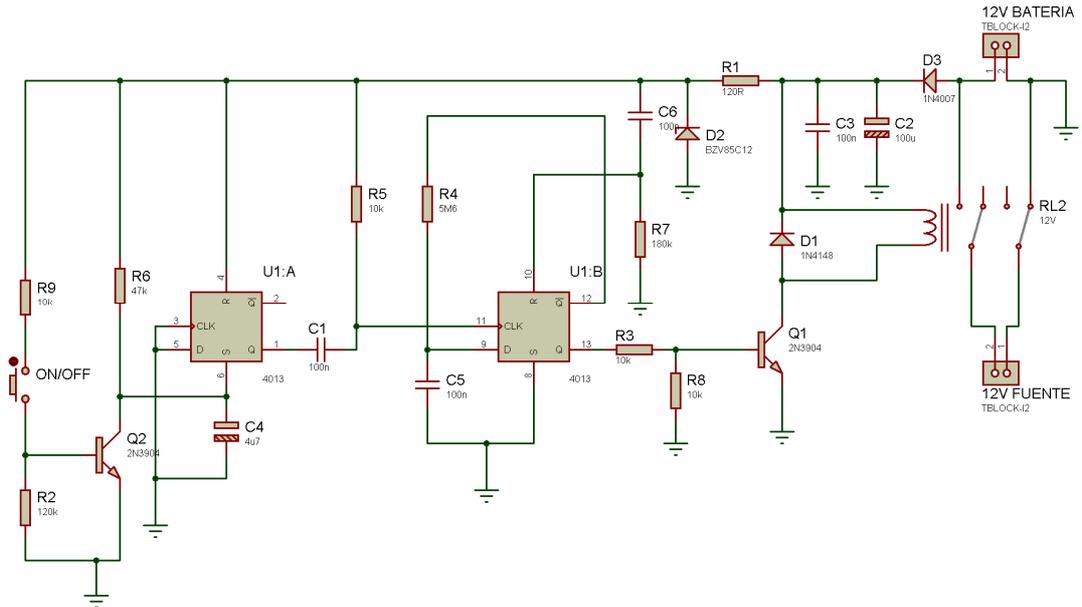


Figura III.6: Diagrama del circuito electrónico para el interruptor electrónico

### 3.4. Adquisición del entorno por ultrasonidos

El módulo que se desarrolló para detectar obstáculos en la ruta del móvil mientras se dirige al destino, está basado en un transmisor receptor de ultrasonidos integrado, que colocados estratégicamente en el frente del robot podrán informar de las condiciones del entorno al sistema en todo el rango frontal lo que permitirá una navegación sin inconvenientes.

#### 3.4.1. Sensor Ultrasonico MaxSonar EZ1

Para la detección de obstáculos se usaron 4 módulos ultrasónicos MaxSonar EZ1 los cuales tienen integrado el transmisor y receptor en la misma placa. Estos sensores son capaces de detectar objetos desde 0 hasta 254 pulgadas (0 a 6.45 metros) y proporcionar una información de salida de la distancia medida en el rango de 6 a 254 pulgadas con una resolución de 1". En el caso de la aplicación propuesta el rango de medición no va más allá de los 3 metros para el funcionamiento correcto del sistema.



Figura III.7: Sensores de ultrasonido MaxSonar-EZ1 usados en el modulo sonar, MAXBOTIX® INC (17).

La medición de distancia con ultrasonido se basa en el principio de “Tiempo de vuelo” (TOF), en donde conociendo la velocidad del sonido en el aire se puede determinar la distancia a un objeto, midiendo el tiempo que demora una ráfaga de ultrasonido en retornar al punto de partida luego de rebotar en dicho objeto, que se explico mas a profundidad en el capitulo anterior “1”.

Para esto es necesario generar esta ráfaga para ser emitida al ambiente y una vez que retorna se debe dar a la señal el tratamiento correspondiente para obtener una medición acertada. Este trabajo es realizado por el sensor ultrasónico MaxSonar-EZ1 gracias a que cuenta con la circuitería interna necesaria y su propio micro-controlador PIC 16F676, la distancia medida se ofrece en tres formatos de salida: anchura de pulso (PW), tensión analógica y salida digital en serie. Gracias a estos tres formatos de salida el sensor MaxSonar-EZ1 se puede conectar a nuestro sistema basado en micro-controlador de una manera fácil y flexible.



cuando se pone a “0” el sistema está detenido. Poniéndola a nivel “1” o simplemente sin conectar, se inicia una nueva medida.

**AN:** Salida analógica de tensión comprendida entre 0 y 2.55 V que representa el valor de la distancia medida. El factor empleado es de 10Mv/pulgada.

**PW:** Este pin proporciona un pulso de salida cuya duración determina la distancia medida. La distancia se puede calcular usando el factor de 147Ms/pulgada.

### **Características de emisión**

Además el sensor MaxSonar-EZ1 cuenta con excelentes características de radiación gracias a que los lóbulos secundarios han sido eliminados por software, reduciendo así la detección de objetos en las llamadas zonas muertas, obteniendo un ángulo de radiación de 36° que para una pared puede llegar a alcanzar los 56°. Los resultados de las pruebas se muestran en una malla de 12 pulgadas para diferentes tamaños del objeto:

- (A) Objeto de 0.25-pulgadas de diámetro, note la emisión estrecha para los objetos pequeños cercanos,
- (B) Objeto de 1-pulgada de diámetro, note el largo patrón de detección estrecho,
- (C) Objeto de 3.25-pulgadas de diámetro, note el largo patrón de detección controlado,
- (D) tabla ancha de 11-pulgadas movida de izquierda a derecha con la tabla paralelo a la cara delantera del sensor y el sensor estacionario.

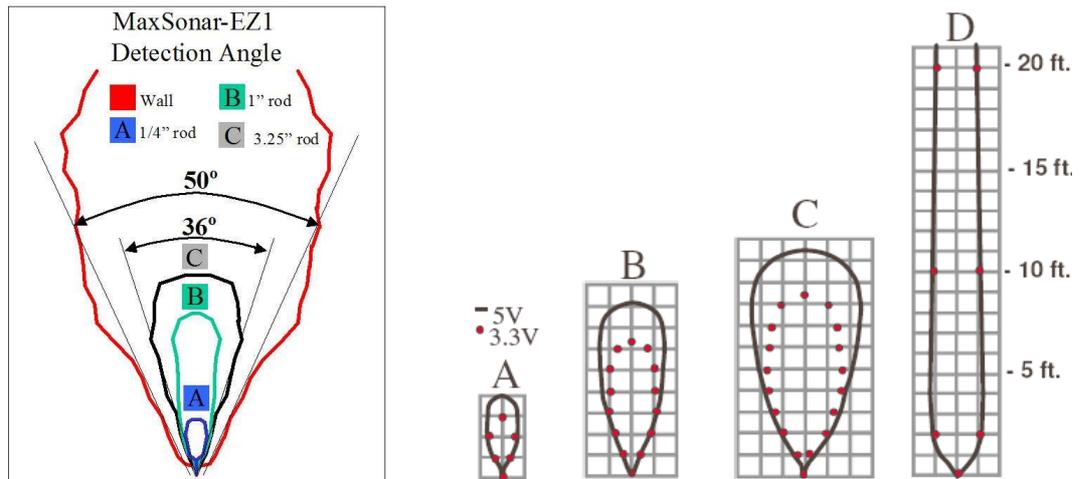


Figura III.9. Capacidad del rango del sensor ultrasónico, MAXBOTIX® INC (17).

### Descripción de Tiempos

A los 250mS después de conectar la alimentación el MaxSonar-EZ1 está listo

para aceptar la señal RX y si está a nivel "1" se ejecuta un ciclo de calibración que dura 49mS y seguidamente comienzan los ciclos de trabajo. Estos consisten en realizar una medida y transmitir la distancia por los tres métodos disponibles. Cada ciclo de trabajo consume 49mS, por lo que el primer ciclo válido tras conectar la alimentación se produce a los 100mS.

Cada ciclo de trabajo de 49mS comienza chequeando el estado de RX. Si está a "0" se da por finalizado ese ciclo. A continuación se transmite una señal ultrasónica de 42KHz y la señal PW se pone a "1". Cuando se detecta un obstáculo PW se pone a "0". Si el tiempo en que esta señal permanece a "1" es superior a 37.5mS significa que no se ha detectado obstáculo alguno. Durante los siguientes 4.7mS se transmite en serie el valor de la medida realizada. El resto del tiempo hasta llegar a los 49mS que dura el ciclo se emplea para ajustar la tensión analógica en AN a su nivel apropiado.

Las características técnicas completas se pueden ver en el anexo 1.

### 3.5. Diseño del módulo ultrasónico detector de obstáculos

Esta tarjeta se diseñó como interfaz entre los sensores ultrasónicos y la tarjeta principal en donde se encuentra ubicado el PIC 16F877, encargado de realizar la recolección de datos de cada uno de los sensores además de proveer de los 5V necesarios para su funcionamiento.

Se utilizaron cuatro sensores ultrasónicos MaxSonar-EZ1 pero, dadas las características del robot y pensando los trabajos futuros se decidió implementar una tarjeta de adquisición para ocho sensores ultrasónicos.

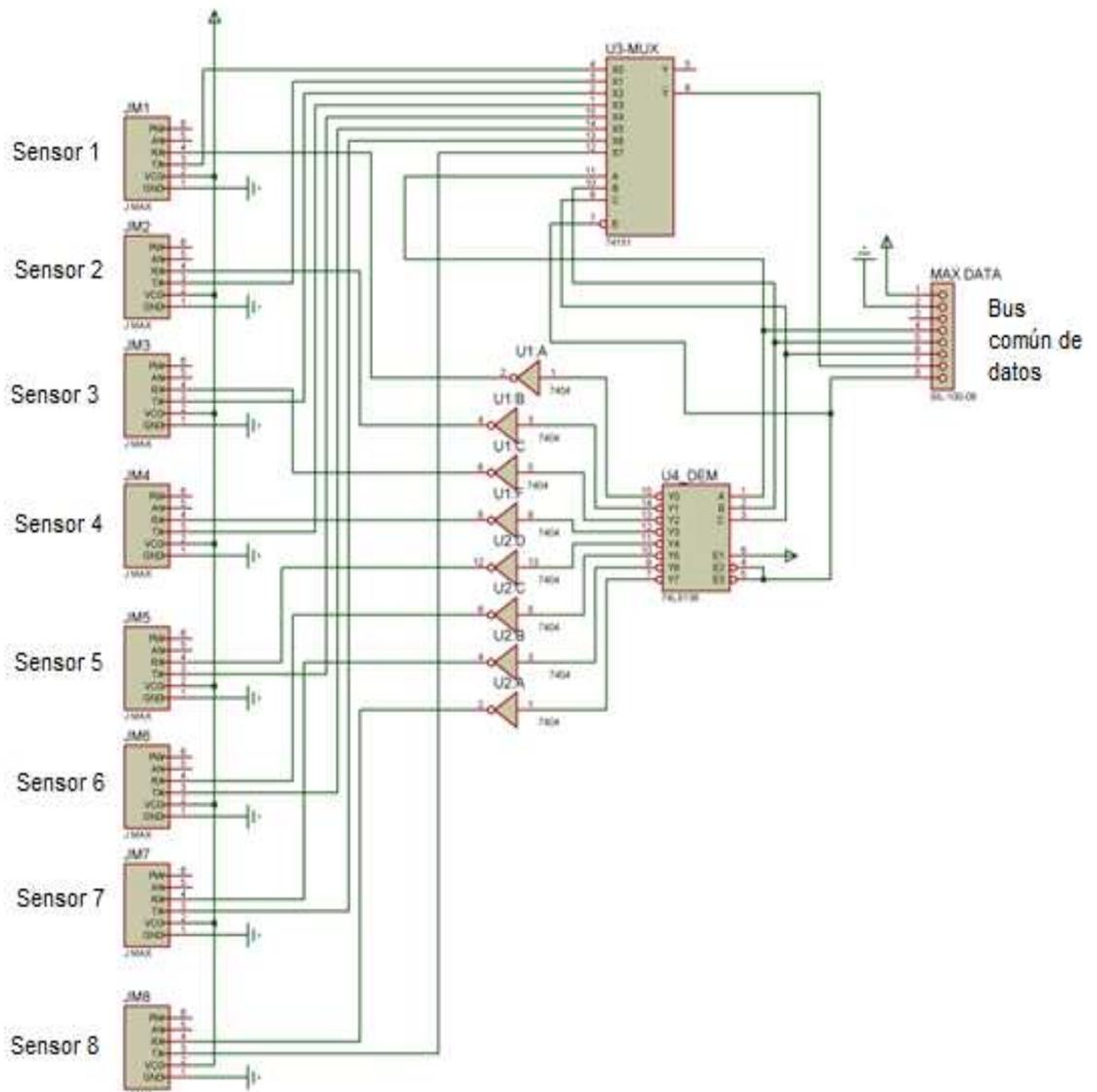


Figura III.10: Circuito de adquisición de datos para los sensores ultrasónicos

Se decidió utilizar la salida de tipo serial debido a que se puede conectar a cualquier dispositivo que contenga una entrada serial tipo USART con transmisión RS232, por su facilidad de implementar en un micro-controlador PIC de gama media o alta.

El diseño se baso en un Multiplexor 74151 y un De multiplexor 74138 los

mismos que cuentan con tres entradas de control y una de habilitación. Se requiere que cada sensor entre en funcionamiento a través de un “1” lógico a la entrada RX.

Para evitar lecturas falsas en los sensores vecinos es fundamental que solo un sensor sea habilitado a la vez hasta que termine su lectura, es aquí donde las entradas de control realizan su trabajo, activando uno por uno cada sensor seleccionando la entrada de datos para la recolección por el PIC.

Cada sensor tiene asociado un código para su selección a través de las entradas de control “A”, “B”, “C” en la tarjeta de adquisición, el programa principal puede seleccionar cualquiera de los 8 sensores para que se realice la medición de distancia a un objeto.

Control			Entrada RX de cada MaxSonar-EZ1							
C	B	A	Max-Ez 0	Max-Ez 1	Max-Ez 2	Max-Ez 3	Max-Ez 4	Max-Ez 5	Max-Ez 6	Max-Ez 7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

Tabla III.1. Activación secuencial de sensores ultrasónicos

Las entradas de control “A”, “B”, “C” están conectadas a los circuitos integrados 74LS151 y 74LS138, que son los encargados de activar y dejar pasar los datos de cada sensor ultrasónico, (Tabla III.1. y III.2).

Existe una entrada de habilitación “E” que se encarga de apagar completamente a todos los sensores ultrasónicos para evitar que alguno permanezca encendido y produzca algún error hasta que el control principal reinicie la lectura.

Control			Salida TX de cada MaxSonar-EZ1							
C	B	A	Max-Ez 0	Max-Ez 1	Max-Ez 2	Max-Ez 3	Max-Ez 4	Max-Ez 5	Max-Ez 6	Max-Ez 7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

Tabla III.2. Control de la salida de datos de cada sensor hacia el micro-controlador

### 3.6. Etapa de control de motores

El sistema de tracción del robot cuenta con dos motores DC que funcionan con un voltaje de 12V, que debido a la carga útil que lleva y por el factor de cambio de velocidad – fuerza llegan a consumir hasta 4 A a toda velocidad. Dadas estas características de potencia, es imposible conectarlos directamente al micro-controlador ya que dañaría irremediabilmente el sistema, es por eso que se utilizó una interfaz de potencia que maneje estos parámetros y permita mantener aislada a la parte de control digital de picos de arranque y señales de ruido provenientes de los motores.

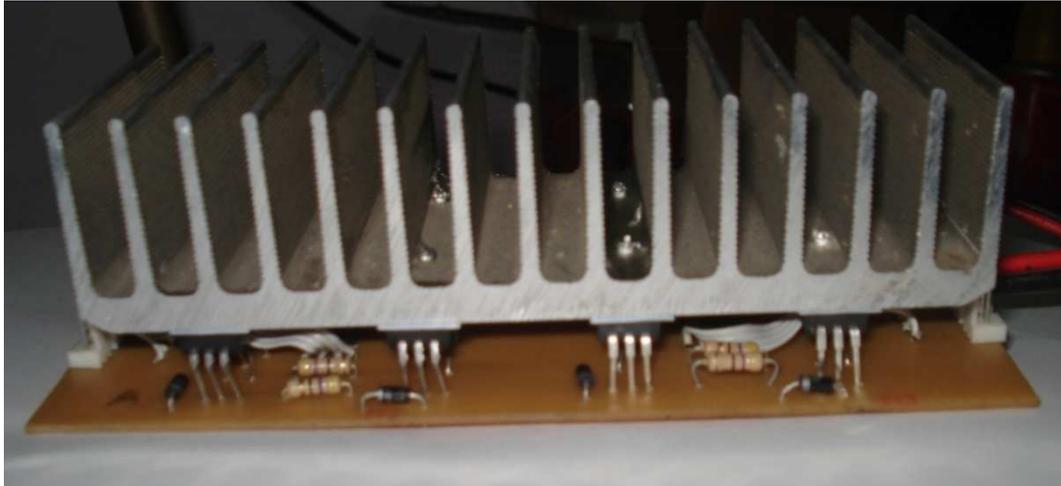


Figura III.11. Controlador de motores DC de potencia

Cada motor debe girar en los dos sentidos y para lograr esto hay que cambiar la polaridad entre sus bornes. No es posible cambiar manualmente estas polaridades por lo que la etapa de potencia debe permitir cambiar el sentido de giro de los motores.

### 3.6.1. Driver de potencia tipo puente H

En el diseño de etapas de potencia para motores es muy común el uso de la configuración en puente H. Esta configuración en su forma más básica consta de cuatro interruptores dispuestos en forma de "H".

Si se accionan los interruptores  $SW_1$  y  $SW_4$  mientras están libres  $SW_2$  y  $SW_3$  se puede activar la dirección del motor hacia adelante, mientras que, si los interruptores  $SW_2$  y  $SW_3$  están activados cuando  $SW_1$  y  $SW_2$  están libres la dirección de reversa del motor es activada, es importante notar que no se puede activar  $SW_1$  y  $SW_3$  o  $SW_2$  y  $SW_4$  a la vez ya que producirían un corto circuito y la destrucción de la fuente de alimentación.

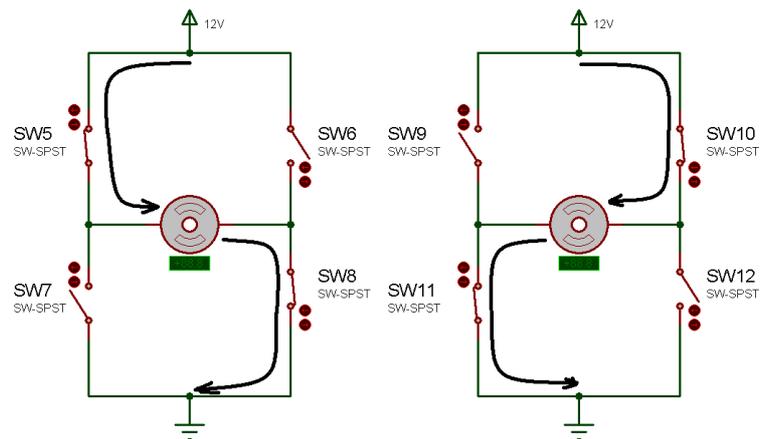


Figura III.12: Configuración básica de un puente H

A este sistema se lo conoce como Puente convertidor completo de cuatro cuadrantes ya que permite controlar el voltaje y la corriente aplicados al motor de forma independiente. El voltaje aplicado al motor esta dado por:

$$V_m = V_{13} - V_{24} \quad (3.1)$$

Donde  $V_{13}$  es controlado por los interruptores  $SW_1$  y  $SW_3$ , y  $V_{24}$  por los interruptores  $SW_2$  y  $SW_4$ .

Por las características del circuito es usual el uso de relés, pero es fácil trasladarlo a transistores bipolares de potencia o MOSFET (transistores de efecto de campo). En la etapa de potencia diseñada se pretendió el uso de transistores MOSFET pero dada la inestabilidad que presentaban y la dificultad para conseguirlos en el mercado local, en especial los de tipo P, se decidió implementar los diseños con transistores bipolares de potencia tipo "P" y "N" Darlington como se puede ver en la figura III.13.

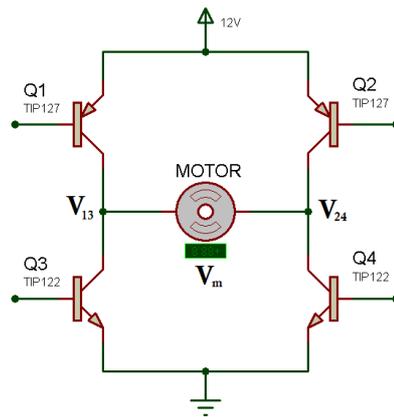


Figura III.13: Puente H con transistores Darlington.

En la configuración se pueden observar los dos tipos de transistor escogidos para facilitar la polarización, se utilizó transistores tipo P y transistores tipo N, de este modo cada transistor funciona como interruptor electrónico y un valor de “1” lógico en la base es requerido para activar los transistores tipo N mientras que un “0” lógico en la base activara los transistores tipo P.

Dado que requerimientos de corriente de cada motor no exceden los 5A, los transistores escogidos fueron los siguientes:

- TIP122: Transistor Darlington tipo N, soporta hasta 5A.
- TIP127: Transistor Darlington tipo P, soporta hasta 5A.

Para usar los transistores como interruptor se los hace un funcionar en " modo de saturación y corte." Un interruptor de transistor puede ser accionado aplicando pequeñas corrientes a su Base, usualmente menores de 20mA. En un transistor NPN cuando una pequeña corriente es aplicada a su base, la corriente fluye de Colector a emisor, en un transistor PNP sucede lo contrario, la corriente fluye de Emisor a colector. Para lograr esto el voltaje aplicado a la Base debe ser más grande que el voltaje de Colector (en un transistor NPN), o más pequeño que el voltaje de Colector (en un transistor PNP). Para asegurar que el voltaje en la base se

encuentre en los niveles deseados se trabaja con los dos tipos de transistores, el PNP con el emisor a 12V mientras el NPN lo tiene a tierra.

La tabla del funcionamiento del puente H, con todas las posibles combinaciones se muestran a continuación:

Transistor Activado				Estados
Q 1	Q 2	Q 3	Q 4	
0	0	1	1	parado
1	1	0	0	parado
1	0	0	1	adelante
0	1	1	0	atrás
1	0	1	0	cortocircuito
0	1	0	1	cortocircuito

Tabla III.3: Funcionamiento del puente H

Como se ve en la tabla existen dos combinaciones que se debe evitar ya que comprometerían la seguridad de todo el sistema y de la batería misma, una solución a este problema es reducir las entradas del puente H mediante circuitos lógicos a únicamente dos, como se muestra en la figura III.14.

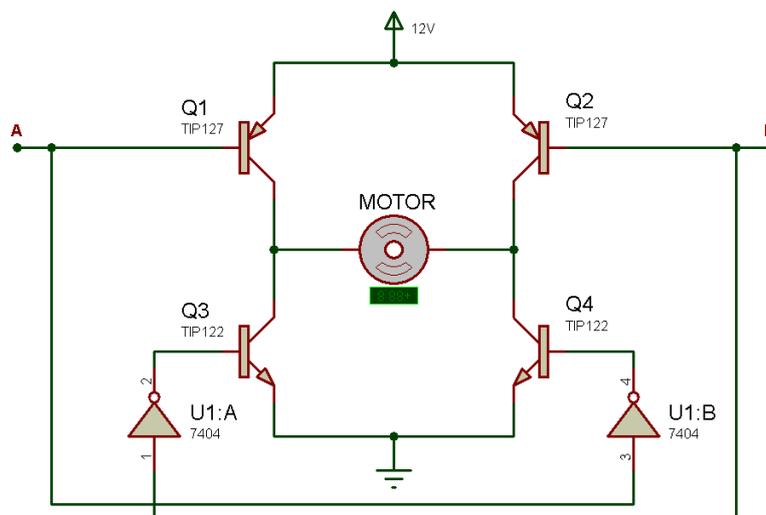


Figura III.14: Puente H con dos entradas de control

Ahora el control del puente H se realiza a través de dos entradas A y B, eliminando el riesgo de un cortocircuito como se muestra en la siguiente tabla.

Entradas De Control		Estados
A	B	
0	0	parado
0	1	adelante
1	0	atrás
1	1	parado

Tabla III.4: Puente H modificado

### 3.6.2. Control de velocidad PWM

Las siglas PWM vienen de la palabra en ingles Pulse Width Modulation o modulación por ancho de pulso. Los drivers de PWM pueden ser usados en una variedad de motores de DC: de campo envolvente o de imán permanente y es afectado por varias características de carga del motor DC. Un motor DC de imán permanente puede ser representado con el circuito equivalente muy simple mostrado en la Figura III.15. Donde  $L_a$  representa la inductancia de armadura,  $R_a$  es la resistencia equivalente en serie, y  $E_a$  la fuerza electromagnética de armadura.

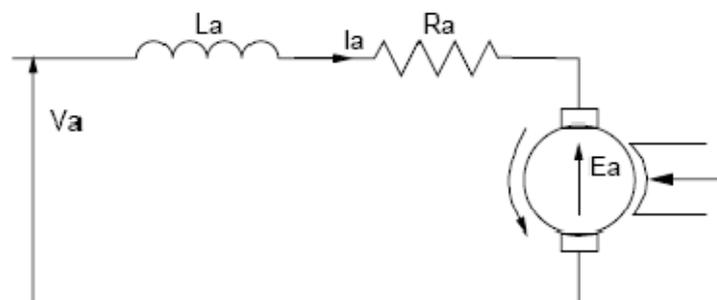


Figura III.15. Circuito equivalente de un motor DC, Humphreys (14).

La fuerza electromagnética inducida representa la porción de la energía a la entrada que es convertida a energía mecánica. La magnitud de fuerza electromagnética es proporcional a la velocidad del motor.

## Principios del sistema de control PWM

La inductancia del motor, puede variar desde unas decenas de uH hasta los mH, tendrá un efecto importante en el diseño del circuito PWM. Debido a que la corriente promedio del motor es una función de la constante eléctrica de tiempo del motor,  $t_a$ , donde:

$$t_a = \frac{L_a}{R_a} \quad (3.2)$$

Para una onda PWM con un periodo de tiempo  $T$  la relación de ancho de pulso para el periodo de encendido-apagado está determinado por  $d$ . El promedio de pulso de corriente dependerá en proporción del ancho de pulso de corriente del motor,  $dT$ , a la constante eléctrica de tiempo del motor,  $t_a$ .

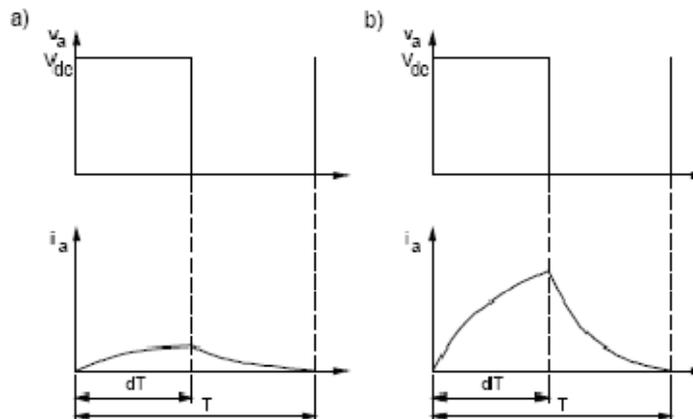


Figura III.16. a) motor DC de alta inductancia  
b) motor DC de baja inductancia, Humphreys (14).

La Figura III.16 muestra las condiciones de dos motores diferentes y un periodo de forma de onda PWM fijado. Para el caso en que la constante de tiempo es mucho más grande que el ancho de pulso, Figura III.16(a) entonces la corriente no puede ser establecida en los enrollados inductivos del motor durante la corta duración del pulso aplicado.

Para un motor de pequeña inductancia y el mismo ancho de pulso, Figura III.16 (b), la corriente de armadura es establecida fácilmente. En la mayoría

de casos un motor que tiene una alta inductancia de armadura requerirá un sistema PWM de baja frecuencia para establecer los niveles de corriente requeridos, y de esta manera el torque necesario. Un motor de baja inductancia permite el uso de altas frecuencias de manera que resulta en sistema de rápida respuesta.

En general, para conseguir una eficiencia óptima en un sistema de control PWM a las mas alta frecuencias practica, el motor debe tener una constante de tiempo,  $t_a$ , cerca de la duración de la forma de onda aplicada  $T$ .

$$t_a = k.T \quad (3.3)$$

donde  $k$  es pequeña.

De manera que una elección apropiada de la frecuencia y la inductancia del motor aseguran una alta corriente del motor durante cada pulso de encendido. Control de corriente del motor, y por tanto control de torque, es lograda al variar el ancho del pulso de la forma de onda aplicada. Como la frecuencia es constante entonces el ancho de pulso retransmite la información de torque del motor.

El torque es dependiente del promedio de corriente del motor (ecuación 3.4) la cual, en cambio, es controlada por el ciclo de trabajo.

$$T_e = k.T.l_a \quad (3.4)$$

Debido a que las características técnicas de los motores usados son desconocidas, el estudio matemático del sistema PWM no pudo ser analizado y se opto por determinar experimentalmente los mejores parámetros de frecuencia para lograr una buena relación de velocidad/torque del motor con el control de velocidad PWM.

Las señales de PWM y sentido de giro son generados por un PIC y para adaptarlas al circuito de potencia puente H, se construyó una etapa lógica mediante compuertas lógicas, del tipo TTL, que se encargan de controlar el modo en que se activan los cuatro transistores del puente H a través de las entradas A y B.

Esta etapa lógica aprovecha los estados que se producen en el puente H como se indicó en la tabla (b de estados con A y B). La tabla final para el diseño es la siguiente:

Salida del pic		Estados	Entradas de control	
Pwm	Direccion		A	B
0	0	PARADO	0	0
0	1	PARADO	0	0
1	0	ATRÁS	1	0
1	1	ADELANTE	0	1

Tabla III.5: Conversión de señales para etapa de PWM

El circuito completo con los circuitos lógicos necesarios y protección es presentado en la figura III.17:

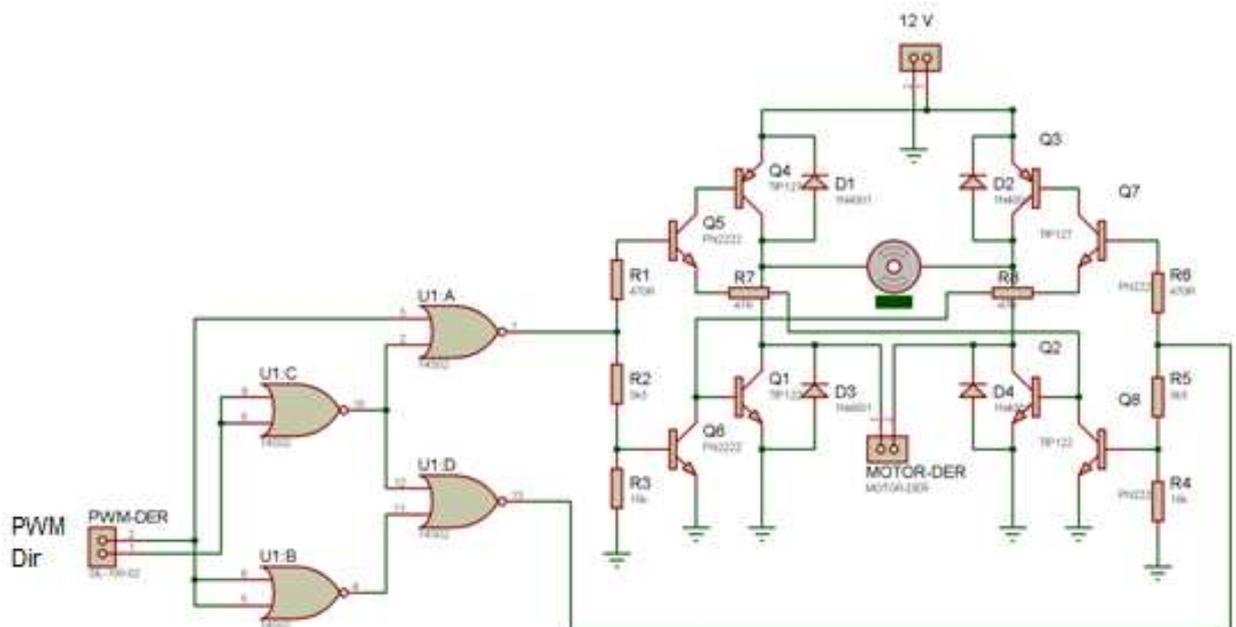


Figura III.17. Esquema electrónico completo del puente H

### 3.7. Brújula digital CMPS03



Figura III.18: Brújula digital CMPS03, SUPER ROBOTICA (29).

La brújula digital CMPS03 es un sensor de campos magnéticos que ofrece una precisión de 3-4 grados y una resolución de decimas. Tiene dos interfaces, mediante pulsos temporizados (modulación en anchura), o bien por medio de un bus I2C, lo que facilita su comunicación con el microcontrolador. Este sensor magnético está específicamente diseñado como sistema de navegación para robots. Está basada en los sensores KMZ51 de Philips que son lo suficientemente sensibles como para captar el campo magnético de la tierra. Usando dos de estos sensores colocados en ángulo de 90 grados, permite al microprocesador calcular la dirección de la componente horizontal del campo magnético natural.

#### 3.7.1. Conexiones y Funcionamiento

En la figura III.19 se puede observar las conexiones de los diferentes pines del CMPS03, entre los que destaca el pin 4 y la 2 - 3, que corresponden a los dos posibles interfaces que incorpora: PWM (Pulse Width Modulation - Modulación por anchura de pulso) y bus I2C formado por las señales SDA (señal de datos) y SCL (señal de reloj) que es un interfaz de comunicación serial bidireccional síncrona.

Norte ←

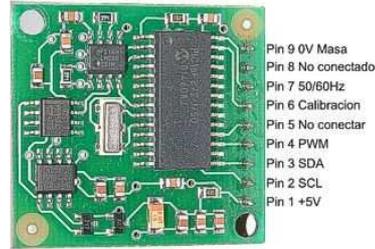


Figura III.19: Pines de conexión de brújula digital CMPS03, SUPER ROBOTICA (29).

En la salida 4, se obtiene una señal PWM en la que el pulso positivo representa el ángulo de la brújula. El pulso varía en duración desde 1ms (0°) hasta 36,99 ms (359,9 °), o dicho de otra forma, el pulso es igual a 100 uS por cada grado más 1ms de tara. La señal permanece a cero durante 65 ms entre pulsos, por lo que el periodo de trabajo es de 65mS + la anchura del pulso. El pulso es generado por un contador de 16 bits del propio procesador, con una resolución de 1 uS.

Otra posibilidad es usar el interfaz I2C formado por los pines 2 y 3, que nos permite una lectura directa del valor en grados de la dirección. El protocolo de comunicación I2C empleado en el modulo, es el mismo que se emplea con la populares eeprom como la 24C04 (Figura III.20). Primero se envía un bit de comienzo, la dirección del modulo (0XC0) con el bit de lectura a cero, y a continuación el numero del registro que se desea leer. Después se vuelve a mandar el bit de comienzo y la dirección del modulo con el bit de lectura a uno (0XC1). Ahora se puede leer uno, o los dos bytes correspondientes a los registros de 8 y 16 bits respectivamente. (El byte de mayor peso se lee primero en los registros de 16 bits).

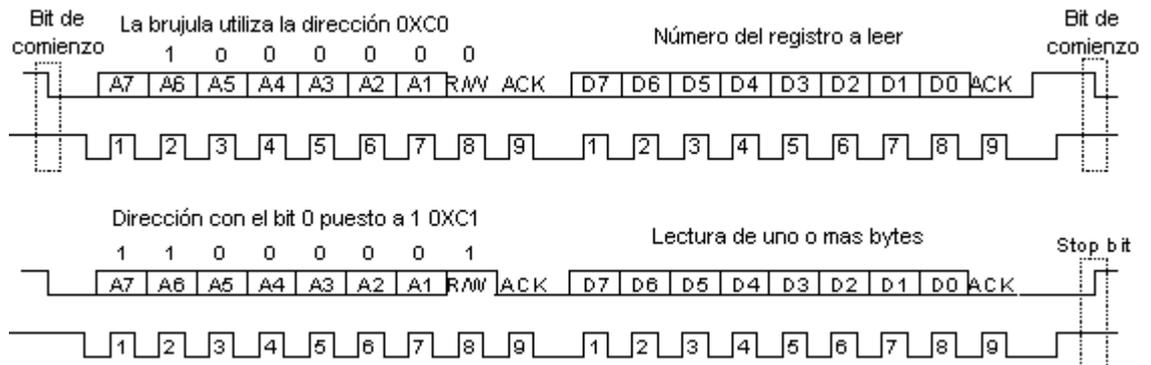


Figura III.20: Funcionamiento del protocolo I2C, SUPER ROBOTICA (29).

El bus I2C del circuito no incorpora las resistencias necesarias de pull-up, por lo que será necesario su implementación externa, para ello es recomendable utilizar dos resistencias de 1.8K en caso de utilizar el bus a 400 KHz y de 1.2K o 1K si se utiliza a una frecuencia de 1Mhz. Pero bajo ninguna circunstancia se podrá prescindir de estas, pues el sistema no funcionara.

El pin 7 se utiliza para seleccionar entre 50 Hz (puesta a cero) o 60 Hz (puesta a uno). Esto es debido a una desviación errónea de unos 1,5° causada por el campo generado por la red eléctrica. Sincronizando la conversión con la frecuencia en hercios de la red, se consigue disminuir el error a tan solo 0,2°. El pin tiene una resistencia interna de pull up, por lo que si se deja sin conectar, funcionara a 60 Hz. El circuito realiza una conversión interna cada 40ms (50 Hz) o cada 33,3 ms (60Hz) de acuerdo con la conexión de esta entrada. No hay ningún tipo de sincronismo entre la realización de la conversión y la salida de los datos, ya que cuando estos son leídos se devuelven el valor más reciente que este almacenado en su respectivo registro.

El pin 6 se usa para calibrar el sensor magnético. Esta entrada tiene su propia resistencia de polarización (pull up) y puede dejarse sin conectar una vez realizada la calibración.

Los pines 5 y 8 están marcados como no conectados, aunque el pin 8 es en realidad el reset del microprocesador, con el fin de poder programarlo una vez soldado al circuito impreso. Para el proyecto no tiene uso ni importancia.

### **3.7.2. Registros de la brújula digital CMPS03**

La brújula tiene un total de 16 bytes de registros, algunos de los cuales forman registros de 2 bytes tal y como se puede ver en la siguiente tabla:

Registro	Función
0	Numero de Revisión del Software
1	Dirección. en 1 byte 0-255 para 0 - 360°
2,3	Dirección. en 2 bytes 0-3599 para 0 - 359,9°
4,5	Test interno señal diferencial sensor 1
6,7	Test interno señal diferencial sensor 2
8,9	Test interno, valor de calibración 1
10,11	Test interno, valor de calibración 2
12	Sin usar, devuelve 0
13	Sin usar, devuelve 0
14	Sin usar, devuelve 0
15	Comando de calibración, escribir 255 para calibrar

Tabla III.6: Registros internos de la brújula digital CMPS03, SUPER ROBOTICA (29).

El registro 0 es la Revisión del software que actualmente es el 8.

El registro 1 es la dirección en grados convertida en un valor entre 0 y 255 para aplicaciones donde resulta complicado utilizar la escala de 0 a 360 grados que requiere dos bytes y que está disponible en los registros 2 y 3 (el 2 es el más significativo) con un valor que va entre 0 y 3599 que equivale a 0 -359,9°.

Los registros 4 a 11 son de uso interno y del 12 al 14 no se usan, por lo que no deberán leerse con el fin de no consumir el ancho de banda del bus I2C.

El registro 15 se usa para calibrar la brújula.

### 3.8. Modulo receptor GPS “SPK-GPS-GS405”

El modulo SPK-GPS-GS405 es un modulo receptor del Sistema de Posición Global basado en una solución de chip de alta sensibilidad SiRF Star III que incluye una antena pasiva omnidireccional Geo-Helix SMP. El módulo receptor puede rastrear simultáneamente 20 satélites, la integración con el receptor GPS actualiza los datos de navegación cada

segundo y provee su la información de localización exacta tan pronto la energía es accionada.

Está diseñado para un amplio espectro de aplicaciones OEM de posicionamiento personal y navegación, como un dispositivo portable o de mano. Cumple con las estrictas necesidades como navegación de autos, mapeo, seguridad entre otras.

### 3.8.1. Características Principales

- Construido con el chipset SiRF Star III con el software estándar SiRF GSW 3.0
- Antena omnidireccional Geo-helix SMP integrada que brinda una alta inmunidad a interferencias de RF.
- Recepción paralela de 20 satélites.
- Soporta el protocolo de datos NMEA0183 V.3.01
- Puerto serial con niveles de salida TTL.

### 3.8.2. Especificaciones técnicas y eléctricas

Es indispensable saber antes de manipular este tipo de artefactos sus características técnicas pues nos ahorra tiempo, se logra una mejor comprensión de su funcionamiento y se lo puede utilizar con mucha más facilidad y confianza.

El módulo GPS cuenta con características eléctricas muy buenas lo que facilita el diseño electrónico como se puede ver en las siguientes tablas:

Parámetro	Min	Typ	Max	Units
Voltaje de alimentación	3.1	3.3	3.5	V
Corriente de operación	67	75	82	mA

Tabla III.7: Valores absolutos máximos de corriente y voltaje, Chung (7).

Parámetro	Min	tipico	Max	Unidad
Frecuencia	1573.42	1575.42	1577.42	Mhz
Ganancia	-5.5	-3	x	dBi
VSWR		2.0:1	2.3:1	
Impedancia		50		ohm
Rango temperatura de operación	-40	20	85	grados C

Tabla III.8: Características de la antena integrada, Chung (7).

	Ítems	Descripción
chipset	GRF3w&GSP3f	tecnología SiRF Star III
general	frecuencia	L1, 1575.42 Mhz
	Código C/A	1.023 Mhz chip rate
	Canales	20

Tabla III.9: Especificaciones técnicas, Chung (7).

Precisión	Posición	10 metros
		5 metros con WAAS habilitado
		< 5 metros (50%) con corrección DGPS
	tiempo	1 us sincronizado al tiempo GPS
Datum	Defecto	WGS 84
	Otros	Selección manual
Tiempo 1er Fix	Adquisición	en promedio 0.1 seg
	inicio instantáneo	en promedio 1 seg
	inicio ambiente caliente	8 seg
	inicio ambiente húmedo	38 seg
	inicio ambiente frio	42 seg
Condiciones dinámicas	Altitud	18000 m
	velocidad	515 m/s
	aceleración	menos de 4g

Tabla III.10: Especificaciones técnicas de precisión. Chung (7).

### 3.8.3. Descripción de los pines

La disposición de pines del modulo GPS es la mostrada en la figura III.21 y su función es descrita a continuación:

**RX:** Este es el canal receptor primario para recibir los comandos del software escrito por el usuario.

**TX:** Este es el canal transmisor primario para la salida de datos de navegación al programa de usuario. La salida de los datos poseen niveles TTL, 0V – 2.85V.

**BATTERY:** Esta es la entrada se utiliza para conectar para una batería de respaldo, sin esta batería el modulo GPS se ejecuta en modo cold star después de ser encendido.

**GPIO1:** El usuario puede usar este pin de entrada/salida para ejecutar funciones especiales como encender un led, que indica cuando una lectura de datos ha sido realizada

**VIN:** Fuente de energía principal de 3.3V con una variación de 5%.

**GND:** Tierra.

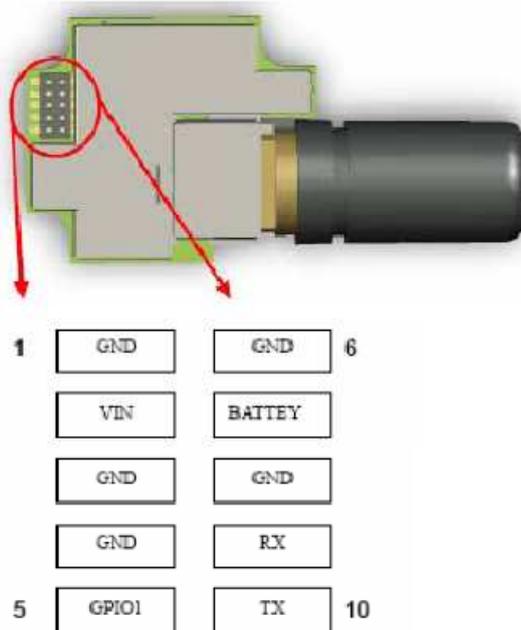


Figura III.21: Disposición de pines para el modulo GPS "SPK-GPS-GS405", Chung (7).

### **3.9. Tarjeta principal**

En el diseño de la tarjeta principal lo primero a tomar en cuenta es dimensionar todas las entradas y salidas que debe contener, para la elección correcta del micro-controlador a usar. Con este objetivo se analizó con detalle cada uno de los módulos, sensores y dispositivos electrónicos empleados:

#### **3.9.1. Control de potencia para motores DC**

El modulo de potencia con puentes H como se analizó en la sección anterior tiene dos entradas de control por cada motor, una para el control de la dirección deseada y la otra para la el control de velocidad mediante la técnica PWM. Por lo que el micro-controlador de la tarjeta de control principal deberá contener estas cuatro salidas para el control de los dos motores de la etapa de tracción del robot.

Las señales de PWM deben ser obtenidas mediante dos módulos CPP, que son pines específicos para trabajar con señales PWM, y las de dirección a través de dos interfaces de salida del micro-controlador.

#### **3.9.2. Sensores ultrasónicos**

Para la adquisición de los datos de los sensores ultrasónicos se diseñó una tarjeta que tiene cuatro entradas de control y una salida de datos. Las cuatro señales de control serán controladas mediante el microprocesador y la entrada de datos debe ser leída utilizando comunicación serial.

#### **3.9.3. Interfaz humana**

Se trata de la interfaz en la que el personal capacitado para manejar el robot, ingresara los datos de la posición de un objetivo, el mismo que el robot deberá alcanzar, es la interfaz de comunicación entre el hombre y el robot, sus principales objetivos son los siguientes:

- Ingresar las coordenadas del objetivo a alcanzar.
- Mostrar los datos de la posición actual.
- Mostrar las lecturas de los sensores ultrasónicos.
- Mostrar los datos de la brújula electrónica

La interfaz humana está compuesta básicamente por un teclado del tipo matricial (figura III.22) y un visualizador LCD con dos líneas de 16 caracteres (figura III.23).

El teclado usa la técnica matricial de 4X3 teclas y son necesarias 7 líneas de E/S del micro-controlador para su total control. Las 7 líneas del teclado se pueden conectar a cualquier puerto del PIC, pero se sugiere conectarlo a las líneas de la puerta B. Esto es porque el mencionado puerto tiene, en la mayoría de los dispositivos PIC, un par de características muy interesantes para el control de un teclado: resistencias pull-up internas para las líneas de entrada y posibilidad de interrupción cuando cualquiera de esas líneas de entrada cambie de estado.

El visualizador LCD, es un potente periférico de salida que va a permitir representar cualquier tipo de mensaje compuesto de letras, números y símbolos produciendo además diferentes efectos de visualización como desplazamientos a izquierda y derecha, parpadeos, deslizamientos verticales, etc. La transferencia de información entre la pantalla LCD y el micro-controlador se realiza en paralelo en grupos de 4 u 8 bits de datos y se conecta con las líneas de E/S.

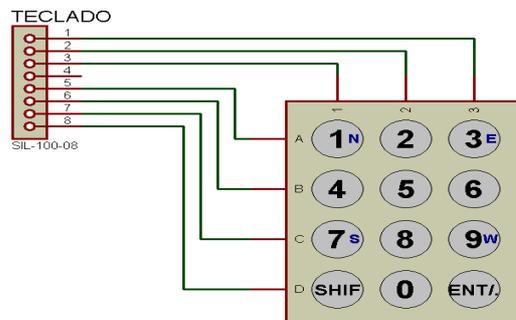


Figura III.22: Teclado matricial 4x3 teclas.

Las señales que controlan el flujo y tipo de información se transfieren a través de las señales de control. La señal RS que determina si la pantalla recibe el código ASCII del carácter a visualizar (nivel "1") o bien un código de instrucción (nivel "0"). Mediante la señal RW el micro-controlador informa a la pantalla LCD si se va a leer o escribir sobre la misma. Finalmente, con la señal E, se habilita o no a esa pantalla. Cuando esta señal vale "0" la pantalla queda desconectada, las líneas de datos quedan en alta impedancia. Cuando vale "1", la pantalla queda habilitada para recibir o enviar datos.

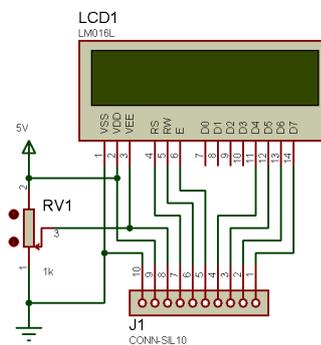


Figura III.23: Visualizador LCD 2 líneas x 16 caracteres.

### 3.10. Microprocesador para el cálculo matemático

Por los requisitos de cálculo matemático que necesitó el robot, fue necesario implementar un co-procesador matemático adicional al PIC del control principal. Este se encargaría de las tareas de cálculo de ángulos y coordenadas actuales, determinación del rumbo, cálculo del vector repulsivo, entre otros.

Al ser todas estas tareas complejas para el micro-controlador, éste debía contar con altas características de memoria y procesamiento, después del análisis y dado que debía encargarse de cálculos matemáticos, mas no de controlar otros periféricos, el numero de pines no era un factor importante, al contrario que la memoria y líneas de código. Tras investigar las características y prestaciones de varios de los micro-controladores PIC de Microchip de la familia 18FXX2 (tabla III.11)

Dispositivo	memoria en el chip		Ram interno	EEPROM
	Flash	# palabras		
PIC18F242	16k	8192	768	256
PIC18F252	32K	16384	1536	256
PIC18F442	16K	8192	768	256
PIC18F452	32K	16384	1536	256

Tabla III.11. Varios modelos prospecto de PIC gama alta, MICROCHIP (20).

Se opto por utilizar el PIC 18F452, un PIC de la familia de gama alta cuyas características se pueden observar a continuación:

Especificaciones	PIC18F242	PIC18F252	PIC18F442	PIC18F452
Frecuencia operación	40 Mhz	40 Mhz	40 Mhz	40 Mhz
memoria programa	16K	32K	16K	32K
memoria instrucciones	8192	16384	8192	16384
memoria datos	768	1536	768	1536
memoria EEPROM	256	256	256	256
interrupciones	17	17	18	18
puertos I/O	A B C	A B C	A B C D E	A B C D E
Timers	4	4	4	4
módulos PWM	2	2	2	2
Comunicación serial	USART	USART	USART	USART
Comunicación paralela			PSP	PSP
modulo ADC 10 bits	5 entradas	5 entradas	8 entradas	8 entradas
Set de instrucciones	75	75	75	75
empaquetado	28 DIP SOIC	28 DIP SOIC	40 DIP PLCC TQFP	40 DIP PLCC TQFP

Tabla III.12. Características técnicas de los PIC 18FXXX, MICROCHIP (20).

El coprocesador matemático utiliza los pines de comunicación serial TX y RX para recibir los datos a procesar y enviar la respuesta de los cálculos, más un puerto para la conexión de un display que sirve para analizar los datos que llegan del micro-controlador de control principal, la figura III.24 muestra el diseño electrónico para este circuito.

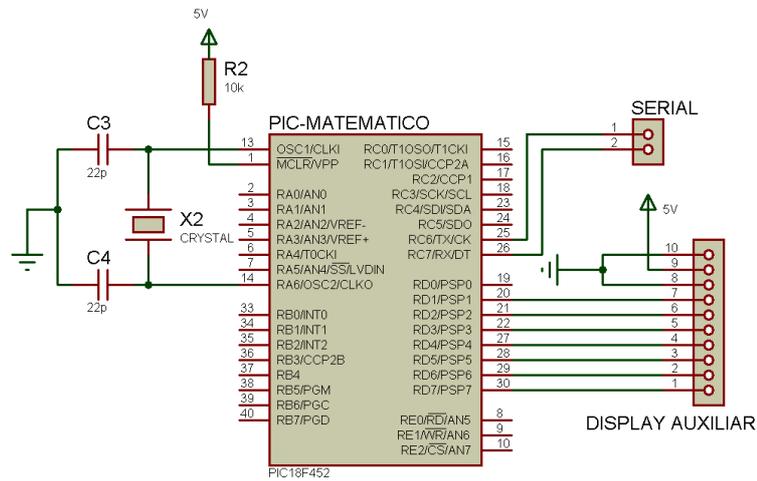


Figura III.24: Diagrama del circuito de procesamiento matemático

Finalmente después de haber realizado el análisis de todos los módulos y dispositivos electrónicos que deben ir conectados a la tarjeta de control principal se realizó un análisis de las necesidades y características que debía cumplir el microprocesador que se resumen en la siguiente tabla.

Módulo	Pines	Dirección de datos	Función
Puente H Izquierdo	DIR	Salida	Control de dirección del motor
	PWM	Salida	control velocidad del motor
Puente H Derecho	DIR	Salida	Control de dirección del motor
	PWM	Salida	control velocidad del motor
Modulo sonar	A	Salida	Control para la selección del sensor ultrasónico
	B	Salida	
	C	Salida	
	E	Salida	Activación modulo sonar
	RX	Entrada	Recepción serial
Teclado	A,B,C, D	Entrada	Escaneo de teclas
	1,2,3	Salida	
Display	RS	Salida	Control del display
	RW	Salida	
	E	Salida	Lectura y escritura de datos
	D7	Entrada/Salida	
	D6	Entrada/Salida	
	D5	Entrada/Salida	
	D4	Entrada/Salida	

Tabla III.13. Requerimientos I/O de la tarjeta principal

Modulo	Pines	Dirección de datos	Función
GPS	RX	Entrada	Recepción de datos GPS con comunicación serial
Brújula digital	SDA	Entrada/Salida	Comunicación I2C
	SCL	Entrada/Salida	Comunicación I2C
Micro-controlador matemático	TX	Salida	Transmisión serial
	RX	Entrada	Recepción serial

Tabla III.13. Requerimientos I/O de la tarjeta principal (continuación)

Dados los requerimientos de entrada y salida de datos así como de características especiales, se decidió trabajar con el PIC 16F877A de microchip, él que cuenta con las siguientes características:

### 3.10.1. Características del núcleo del micro-controlador

- CPU de altas prestaciones, con estructura RISC
- Sólo 35 instrucciones de una sola palabra
- Todas las instrucciones duran un ciclo de reloj, menos las de salto, que duran 2
- Soportan un reloj de hasta 20Mhz
- Hasta 8K x 14 palabras de memoria de programa FLASH
- Hasta 368 x 8 bytes de memoria de datos RAM
- Hasta 256 x 8 bytes de memoria de datos EEPROM
- Compatible pin a pin con los dispositivos: PIC 16C73B/74B/76/77
- Hasta 15 fuentes de interrupción distintas
- Pila hardware de ocho niveles
- Modos de direccionamiento directo, indirecto y relativo
- Reset Power-on
- Temporizador Power-up y oscilador Start-up
- Perro guardián (WDT) con su propio oscilador RC

- Protección de código
- Modo SLEEP de ahorro de energía
- Posibilidad de selección entre distintos osciladores
- Programación serie en el circuito mediante dos pines
- Depuración del código en el circuito mediante dos pines
- Funcionamiento en un gran rango de voltaje: 2.0V a 5.5V
- Fuentes / sumideros de gran corriente: 25mA
- Bajo consumo

### **3.10.2. Características de los periféricos**

- Timer0: Temporizador / contador con prescalador de 8 bits
- Timer1: Temporizador /contador de 16 bits, que se puede incrementar durante el estado de SLEEP (bajo consumo) mediante un oscilador externo
- Timer2: Temporizador contador de 8 bits con registro de periodo de 8 bits, prescalador y postescalador
- Dos módulos de captura, comparación y PWM. PWM de resolución máxima de 10 bits
- Convertidor analógico – digital de 10 bits, multicanal
- Puerto serie síncrono (SSP) con SPI e I2C
- USART / SCI con detección de dirección de 9 bits
- Puertos de entrada A,B,C,D,E

El diagrama del circuito de la tarjeta de control principal con todas las conexiones necesarias es mostrado en la figura III.25.

Tal como se puede las conexiones para el modulo sonar se realizaron al puerto A del PIC, para el control del teclado se conecto al puerto B, el display está controlado por el puerto D, para la comunicación serial tanto del PIC de cálculo matemático y del modulo receptor GPS, PWM de control de velocidad y comunicación I2C de la brújula digital se utilizó el puerto C

dado que este cuenta con las características especiales de hardware para todas estas tareas como son las líneas de transmisión USART, puerto I2C y dos módulos CPP para control PWM.

Los detalles de las conexiones se podrán aclarar en la figura III.25 donde se encuentra el circuito electrónico del sistema completo.

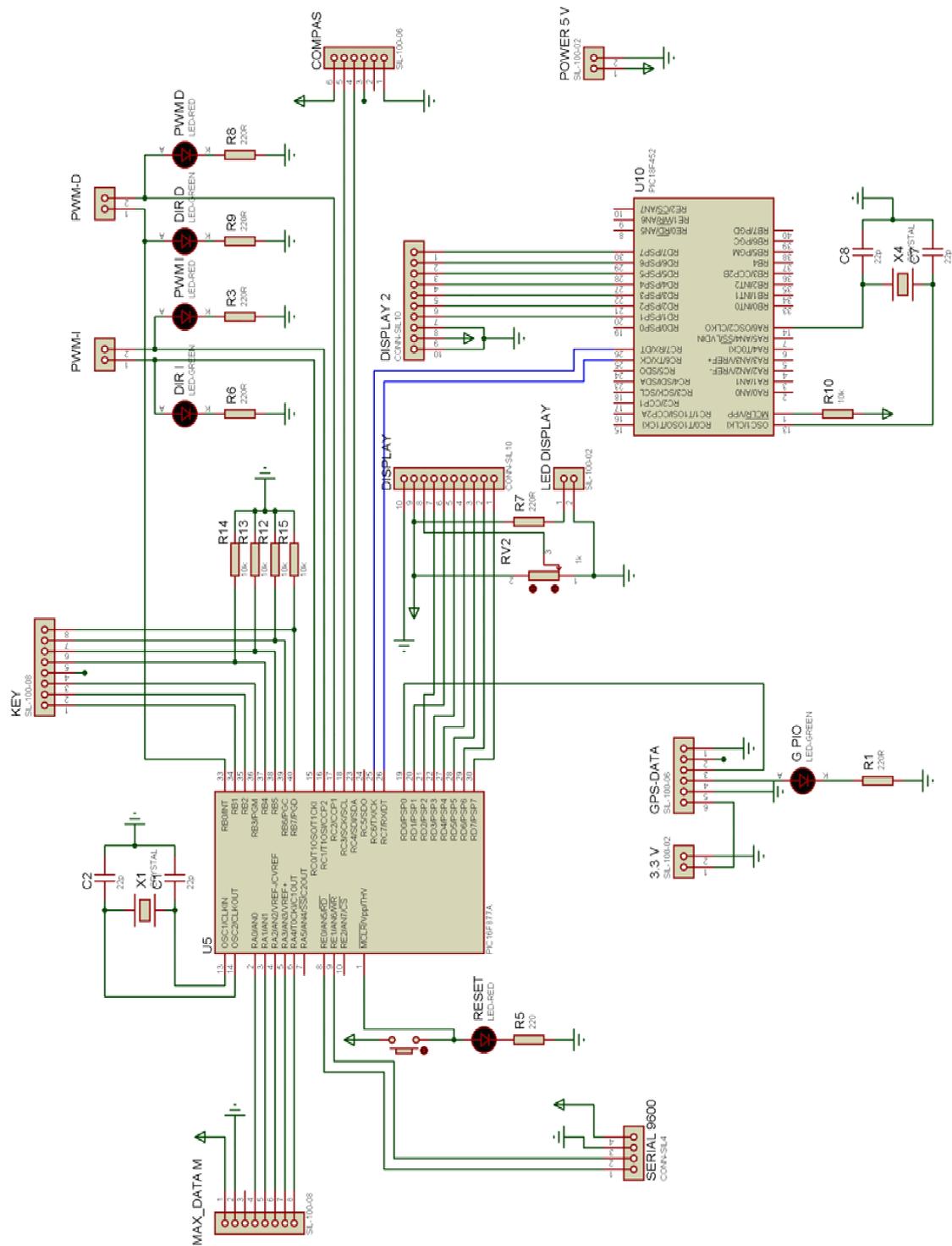


Figura III.25. Diagrama electrónico de la tarjeta de control principal

## **CAPITULO IV**

### **SOFTWARE**

La programación de los microcontroladores, de control principal (PIC 16F877A) como del pic matemático (PIC 18F452) se desarrolló en lenguaje BASIC, bajo el entorno de Microcode Studio y Mikrobasic respectivamente. El uso de este lenguaje fue fundamental, dado que es un lenguaje de programación de alto nivel, mucho más simple que el ensamblador. En el presente capítulo se explica todas las consideraciones tomadas para el desarrollo del control del robot móvil.

#### **4.1. Metodología**

Como metodología de desarrollo se ha optado por una metodología estructural basada en la propuesta realizada por MARTÍN (15), la cual se adapta al desarrollo de programas para micro - controladores. El modelo de desarrollo utilizado es el incremental, el software cuenta con varias entradas de sensores que añaden nuevas funcionalidades de control, las cuales deben ser añadidas de forma incremental siguiendo la metodología hasta lograr el sistema completo.

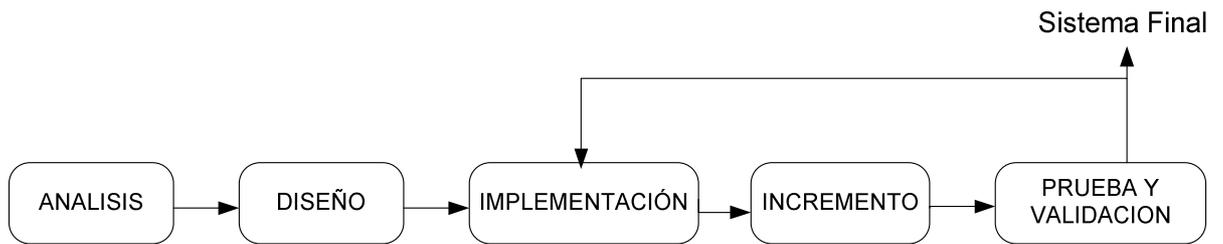


Figura IV.1: Metodología de desarrollo: Modelo incremental

## 4.2. Análisis

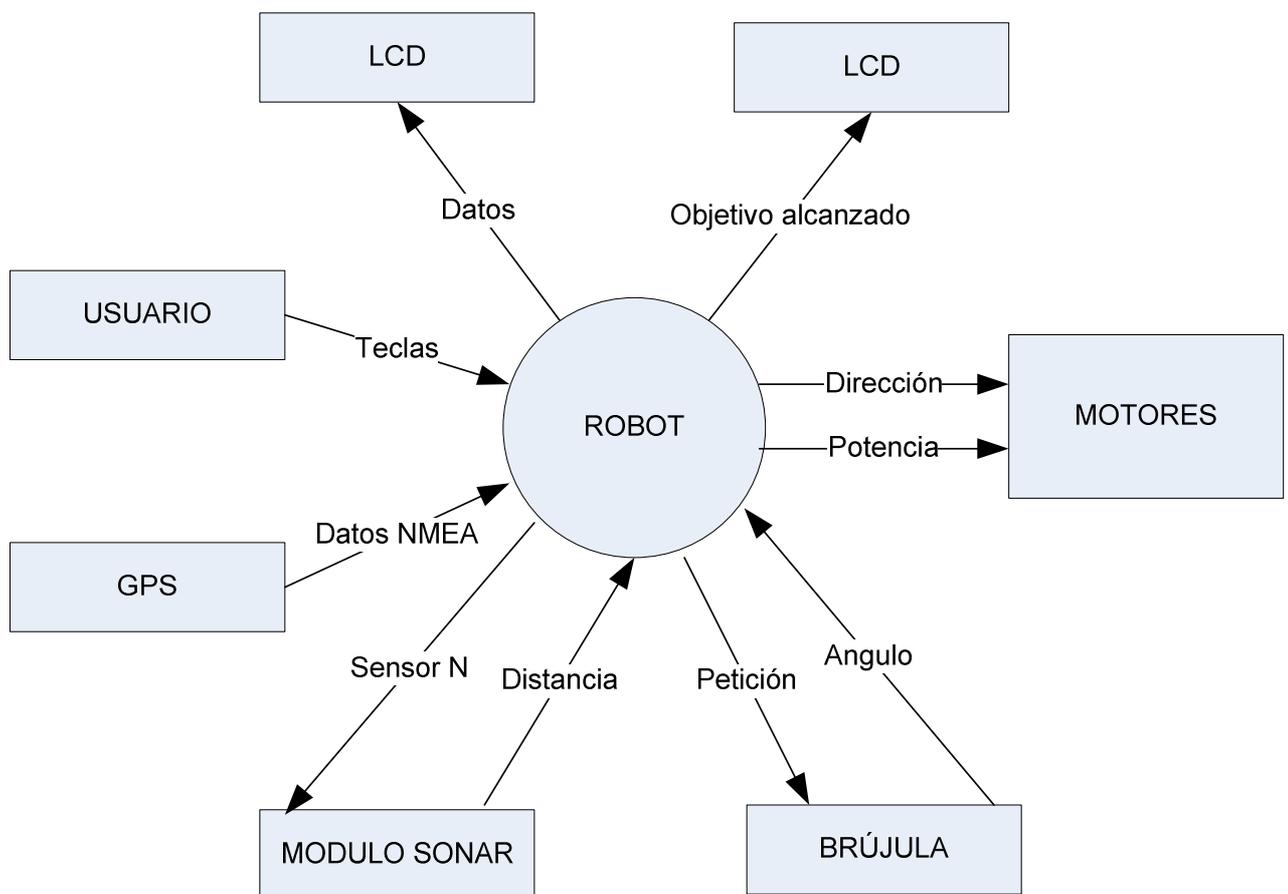


Figura IV.2. Nivel 0: DFD Diagrama de contexto del Sistema

Como muestra la figura IV.2, en forma global el sistema tiene que interactuar con cada uno de los sensores y con el usuario que es la persona encargada de ingresar las ordenes al robot. La comunicación del sistema con los sensores y con el usuario genera el movimiento del robot hacia un punto (objetivo), que es la razón principal del presente proyecto.

### 4.2.1. Requisitos

El desarrollo del proyecto está fundamentado en los objetivos generales y específicos comentados al inicio de este proyecto y debe ajustarse a los siguientes requisitos:

- El desarrollo del sistema de control del robot estará basado en Basic, un lenguaje de alto nivel bajo los entornos de programación Microcode y MicroBasic 8.0.
- Las pruebas y validación de cada incremento serán realizadas bajo el entorno de simulación Proteus 7.0 configurando cada una de las variables para que se asemeje a la realidad.
- El sistema deberá ser capaz de establecer la comunicación e interpretación de datos de cada uno de los sensores, que primordialmente lo hacen en forma serial asincrónica y con formato de datos ASCII.
- El programa deberá ser capaz de controlar los motores del sistema robótico, generando movimientos básicos como avanzar, retroceder, girar y parar.
- El programa deberá también controlar los niveles de potencia suministrados a los motores basándose en técnicas PWM por software.
- El programa debe manejar una interfaz de usuario que permita el ingreso del destino objetivo además de mostrar los datos obtenidos por los sensores y los cálculos realizados.
- Debe contener todas las instrucciones para realizar los cálculos de rumbo o dirección al objetivo además debe ser capaz de reconocer cuando ha alcanzado el objetivo.
- Debe contener un algoritmo que basado en las lecturas de los sensores ultrasónicos permita reconocer un obstáculo y evitarlo en forma parcial o total dependiendo de las limitaciones propias del hardware.

De los requisitos encontrados se pudo desarrollar un primer análisis de los posibles procesos y datos que se debía seguir para la consecución del proyecto, la subdivisión está basada en el hecho de que cada sensor necesita un proceso que interprete los datos para que estén listos para un posterior análisis o calculo, en la figura IV.3 se muestra esta subdivisión en un diagrama de flujo de datos de nivel 1.

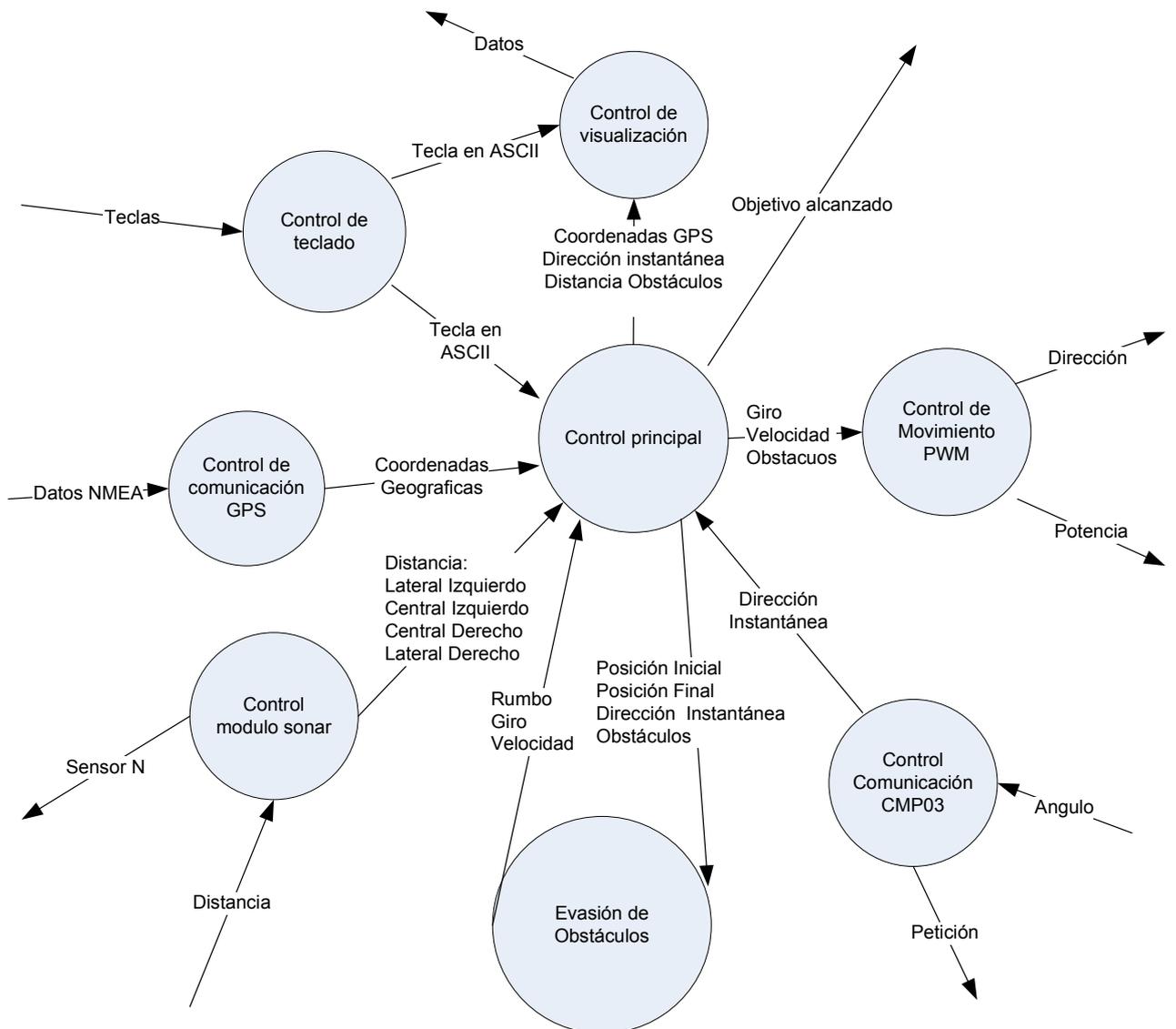


Figura IV.3. Nivel 1: DFD Diagrama del Sistema Robótico

## 4.3. Diseño

### 4.3.1. Control de teclado

El teclado no es más que una matriz de pulsadores que tienen asociado un carácter ASCII, que para nuestro caso son los números del 0 al 9 el punto decimal mas cuatro letras que indican los cuatro puntos cardinales N(Norte), S(Sur), E(Este) y W(Oeste) dados los estándares mundiales. Además cuenta con seis teclas de función que tienen asociados la función enter, shift mas las teclas para mover el cursor. Las teclas de puntos cardinales, de cursor y enter son activadas únicamente después de presionar la tecla shift.

La principal función de la tecla enter es para hacerle saber al programa principal, que se ha ingresado las coordenadas del objetivo y que proceda a tomar control del robot para llevarlo al objetivo. Las demás teclas trabajan directamente con el Display LCD de manera que muestra las coordenadas escritas así como la navegación dentro del display para poder editar el texto, el algoritmo para la lectura de las coordenadas del objetivo (destino) se muestra en la figura IV.4.

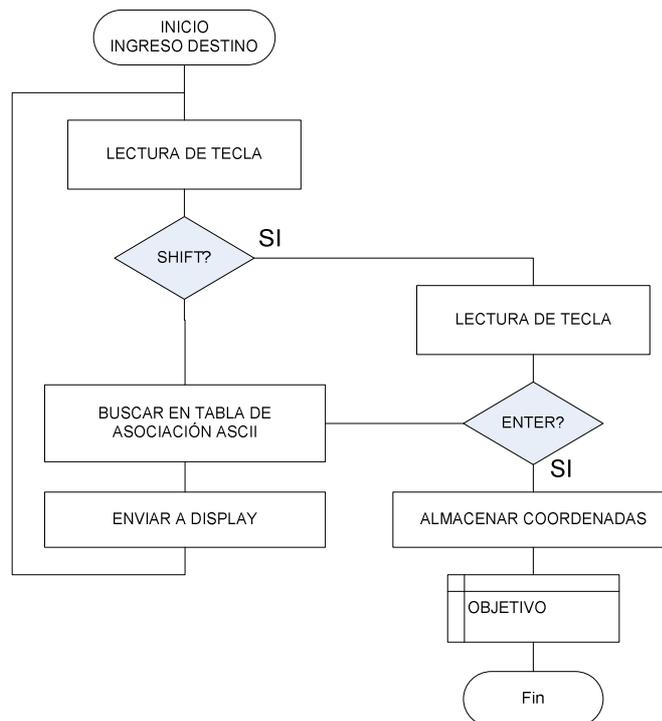


Figura IV.4: Algoritmo de ingreso de coordenadas del objetivo

A mas de ser la interfaz por donde se ingresa las coordenadas del objetivo el teclado también sirve para seleccionar la lectura de cada uno de los módulos y sensores para lo cual al inicio de la ejecución del programa se puede seleccionar mediante la tecla uno, cuatro, siete la lectura del modulo GPS, brújula y modulo sonar respectivamente, este proceso es realizado como se indica en la figura IV.5.

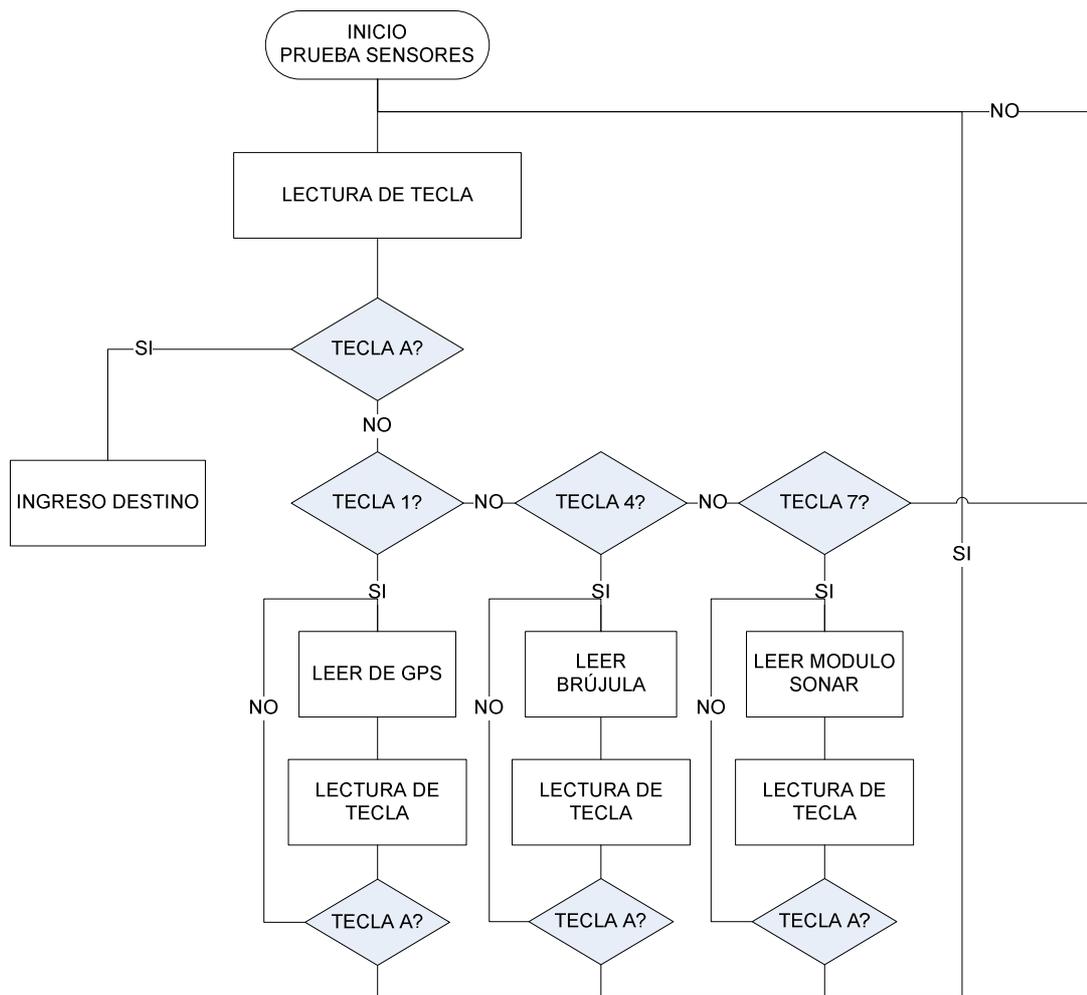


Figura IV.5: Algoritmo de prueba de sensores

### 4.3.2. Comunicación con el módulo receptor GPS

Como se revisó en el capítulo 1 existen una gran variedad de sentencias NMEA para diferentes equipos y para diferentes fabricantes. El módulo receptor GPS usado, envía en forma autónoma y continua, cuatro sentencias NMEA diferentes que se muestran en la tabla IV.1.

MENSAJES	DESCRIPCIÓN
<b>GGA</b>	Datos de la posición global
<b>GSA</b>	GNSS DOP y satélites activos
<b>GSV</b>	Satélites GNSS a la vista
<b>RMC</b>	Datos mínimos recomendados GNSS

Tabla IV.1.: Mensajes NMEA entregados por el modulo receptor GPS

Estos mensajes contienen varios datos de los cuales, los más importantes para nuestro proyecto son GGA y RMC, ya que contienen la posición exacta del módulo receptor en el globo terrestre, mostrados en forma de coordenadas geográficas de Latitud y Longitud, así como información de los satélites entre otros. Antes de poder utilizarlos, estos datos deben ser extraídos de todo el mensaje y para ello se debe analizar la forma en que se los reciben.

Cada mensaje NMEA es transmitido en formato ASCII empieza con un "\$" seguido de GP y a continuación el mensaje enviado con cada parámetro separado por comas como se muestra en la figura IV.6.

\$GPGGA,161229.487,3723.2475,N,12158.3416,W,1,07,1.0,9.0,M,, , ,0000\*18

Figura IV.6. Mensaje GGA de NMEA, el área encerrada corresponde a las coordenadas geográficas de un punto.

Con todos estos datos es posible extraer los parámetros deseados que en nuestro caso son los datos de posición (Latitud, Longitud). El diagrama para la extracción de los datos es el siguiente:

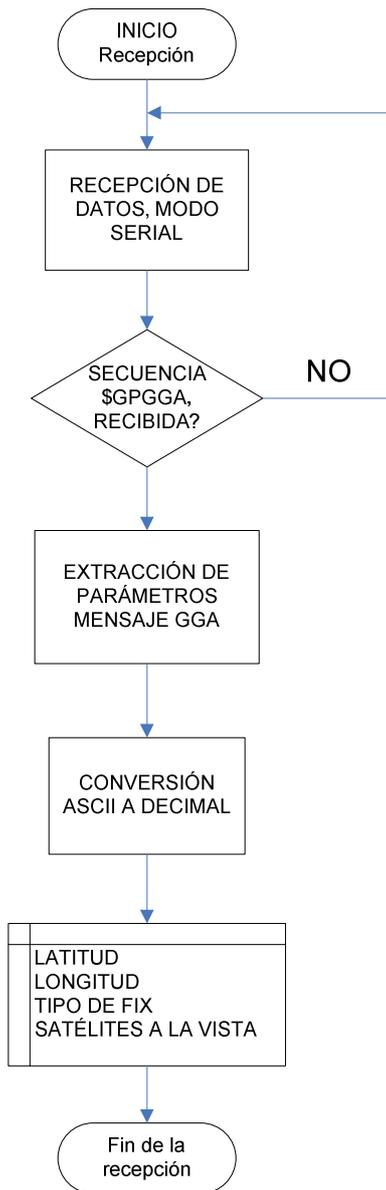


Figura IV.7: Algoritmo de lectura de datos del módulo GPS

Después que el mensaje en formato ASCII ha sido almacenado un vector que se define como  $GPS\_RX[n]$ , cada uno de los datos de cada parámetro son extraídos, se realiza una conversión de formato ASCII a decimal y se los almacena en su respectiva variable.

### 4.3.3. Control módulo sonar

El módulo sonar está constituido inicialmente por cuatro sensores ultrasónicos, con la posibilidad de ser ampliado a ocho. Por las características de las ondas ultrasónicas, las lecturas de cada sensor están sujetas a múltiples errores. Los más frecuentes son la pérdida del eco, el crosstalk y los falsos ecos.

La alternativa de solución para mejorar el sistema y minimizar al máximo esos errores es la lectura en forma secuencial de cada uno de los sensores ultrasónicos que componen el módulo sonar, además de implementar un algoritmo de promedio aplicado a 5 lecturas diferentes de cada sensor. Es decir se enciende únicamente un sensor ultrasónico mientras los demás continúan apagados, las medidas a un objeto son tomadas y promediadas.

Las tareas a ser ejecutadas en el modulo sonar para medir los obstáculos en el entorno son mostradas en la figura IV.8. El resultado de las mediciones realizadas es almacenado en cuatro variables dentro del PIC16F877, **MAX\_LI**, **MAX\_LD**, **MAX\_CI**, **MAX\_CD**, las que corresponden a los sensores ubicados al extremo lateral izquierdo, lateral derecho, central izquierdo y central derecho respectivamente que posteriormente pueden ser utilizados para el análisis del entorno que rodea al robot móvil.

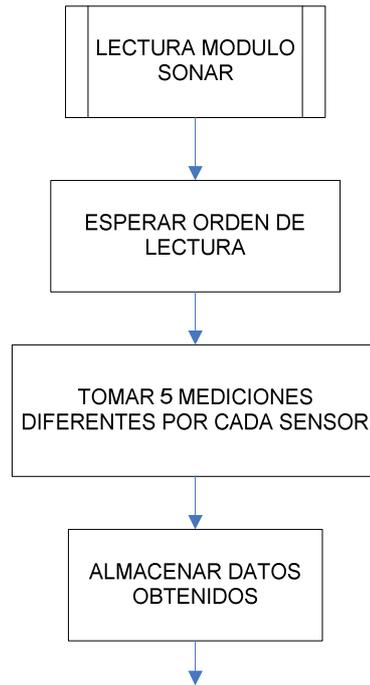


Figura IV.8. Tareas ejecutadas para el módulo sonar

El programa espera la orden de lectura del entorno, a continuación envía la orden de lectura que indica que sensor debe tomar la lectura del entorno en forma secuencial hasta que se haya tomado las cinco mediciones de cada sensor.

#### 4.3.4. Algoritmo de promedio

El principal objetivo del algoritmo de promedio es eliminar al máximo los errores generados por los falsos ecos y el crosstalk, 5 mediciones por cada sensor son realizadas y después promediadas aritméticamente como en el ejemplo mostrado en la Tabla IV.2.

	Medida 1	Medida 2	Medida 3	Medida 4	Medida 5	Promedio
Sensor 1	65	6	63	53	68	<b>51</b>
Sensor 2	6	200	190	6	6	<b>81,6</b>
Sensor 3	120	6	6	6	6	<b>28,8</b>
Sensor 4	47	50	48	45	43	<b>46,6</b>

Tabla IV.2: Promedio algebraico para 5 medidas realizadas por cada sensor

Los valores de la medición realizada por cada sensor pueden variar desde un valor mínimo de 6 hasta un máximo de 254 que representan directamente la distancia a un objeto en pulgadas, todos los objetos que se encuentren muy próximos al sensor ultrasónico a una distancia entre 0 y 6 pulgadas tienen como respuesta del sensor un valor de 6, valor que hay que tomar en cuenta a la hora de realizar los cálculos.

De los resultados obtenidos en la tabla IV.2. para el sensor 3, que es el peor de los casos, el promedio arroja una distancia aproximada de 28 pulgadas hacia el obstáculo, pero cuatro medidas de las cinco que se han tomado arrojan un valor de 6 lo que indica claramente la presencia de un obstáculo en las proximidades del robot, esta lectura errónea provoca que el robot avance y choque directamente con el obstáculo.

Para evitar estos problemas un análisis de los datos de las cinco mediciones es realizado, si existen más de dos medidas iguales a 6 el valor del promedio es igual a 6 de lo contrario se promedian solo las lecturas diferentes de 6, el resultado de este análisis se muestra en la tabla IV.3. Ahora para el sensor 3 la respuesta del promedio indica claramente la presencia de un obstáculo en las proximidades inmediatas del robot de esta manera se logra reducir el peligro de colisión debido a los falsos ecos.

	Medida 1	Medida 2	Medida 3	Medida 4	Medida 5	Promedio
Sensor 1	65	6	63	53	68	62,25
Sensor 2	6	200	190	6	6	6
Sensor 3	120	6	6	6	6	6
Sensor 4	47	50	48	45	43	46,6

Tabla IV.3: Promedio aritmético revisado

El algoritmo de promedio selecciona el sensor del cual se toman 5 medidas mediante las entradas A, B, C que controlan el MUX y el DEMUX del módulo sonar, además para evitar que alguno de los sensores continúe emitiendo ondas ultrasónicas sin que el programa de control principal lo

solicite, cuenta con una entrada de habilitación E que apaga completamente el modulo sonar, el diagrama para el algoritmo de promedio es mostrado a continuación

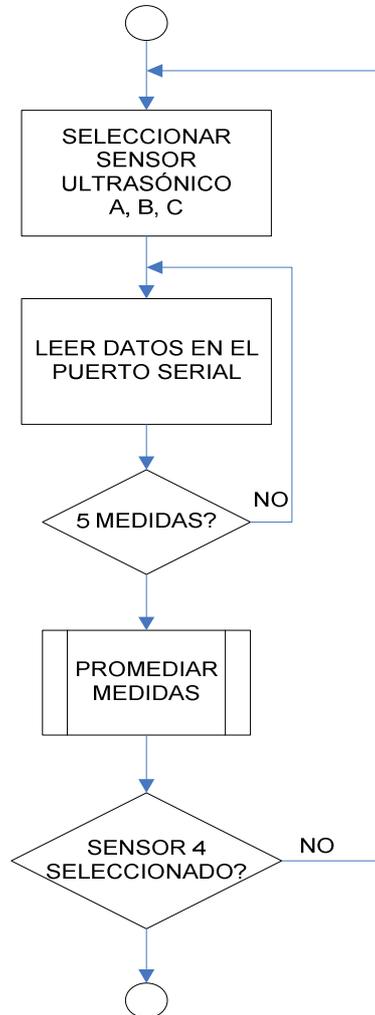


Figura IV.9: Algoritmo de lectura y promedio

#### 4.3.5. Comunicación del módulo sonar

Cada uno de los sensores ultrasónicos que conforman el modulo sonar realizan una transmisión serial de los datos de la distancia obtenida a un objeto u obstáculo, esta transmisión se realiza bajo los parámetros mostrados en la tabla IV.4.

Se transmiten 5 bytes de datos por cada medida realizada. Empieza con el carácter ASCII "R", continua con tres caracteres ASCII con los dígitos de la medida propiamente dicha y comprendido entre 006 y 254 y finaliza con el código de retorno de carro (0x0D).

PARÁMETRO	ESPECIFICACIÓN
Interface	Simple UART
Bit rate	9600 bps
Start bit	1 bit
Stop bit	1 bit
Data bit	8 bit
Parity	None
Transmission data	NMEA 0183 Ver3.01

Tabla IV.4: Parámetros de comunicación con el modulo sonar

El algoritmo para la lectura de un sensor ultrasónico se muestra en la figura IV.10. El programa espera que llegue el carácter ASCII "R" y después lee los tres caracteres siguientes que representan la distancia medida por el sensor seleccionado.

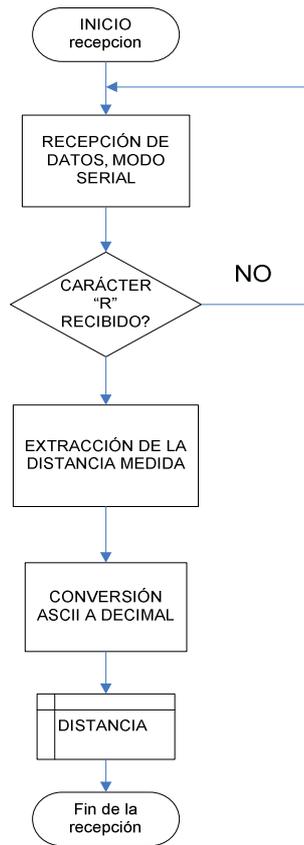


Figura IV.10: Algoritmo de lectura de datos para un sensor ultrasónico

#### 4.3.6. Comunicación de la brújula electrónica

La comunicación con la brújula digital es realizada a través de un canal de comunicación I2C, la brújula se encuentra continuamente en modo esclavo, así que el PIC 16F877 se encuentra en modo maestro al realizar la comunicación.

El algoritmo para la lectura de la brújula se muestra en la figura IV.11. El programa envía la orden de lectura a la brújula, a continuación se envía el registro y el número de bytes a ser leídos, la brújula responde enviando los datos solicitados que no es más que la dirección instantánea en forma de ángulo.

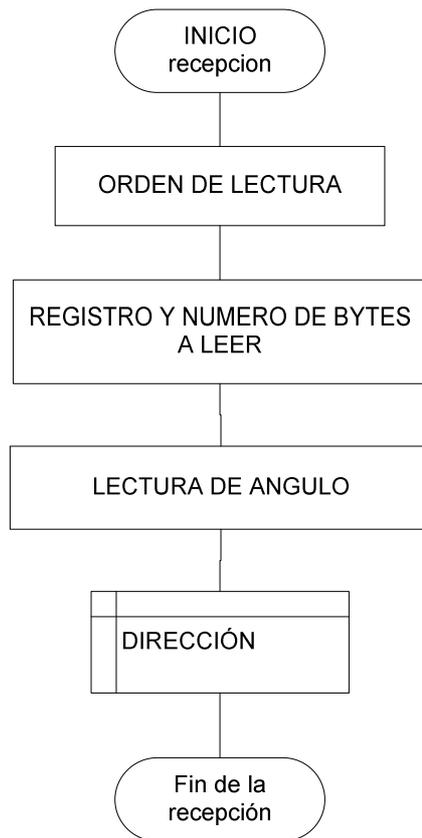


Figura IV.11: Algoritmo de lectura de datos para la brújula electrónica

## 4.4. Integración del programa principal

### 4.4.1. Navegación por satélite

Este tipo de navegación es práctico y ágil, el móvil se apoya en el GPS y la brújula para realizarlo. Se ingresa al sistema un punto de objetivo en coordenadas geográficas que forman una trayectoria, el programa principal lee estos datos y los almacena en memoria, este punto será buscado y encontrado por el móvil.

Para encontrar el primer punto el robot precisa de su ubicación actual la que obtiene con ayuda del GPS, procesa esta información y la compara con el punto a encontrar, con ayuda de la brújula se determina la dirección a seguir y se ubica en esta dirección para después avanzar en busca del punto, a medida que avanza lee periódicamente el GPS y la brújula con tal

de asegurarse de ir en el sentido correcto, de no ser así toma las correcciones del caso hasta llegar al punto encomendado, al momento que encuentra el punto comparando las coordenadas dadas con las generadas por el GPS, claro está con un rango de error de 10 m máximo, medida impuesta por el error de localización del GPS utilizado, se cumple el objetivo y se exponen los resultados satisfactorios del proyecto.

De existir obstáculos en la trayectoria el móvil acudirá al algoritmo de evasión de obstáculos VFF, que se apoya en la detección de obstáculos por medio del módulo sonar.

El algoritmo de navegación utiliza todos los análisis hechos anteriormente. El proceso que realiza el robot en general para la navegación GPS con evasión de obstáculos es mostrado a continuación:

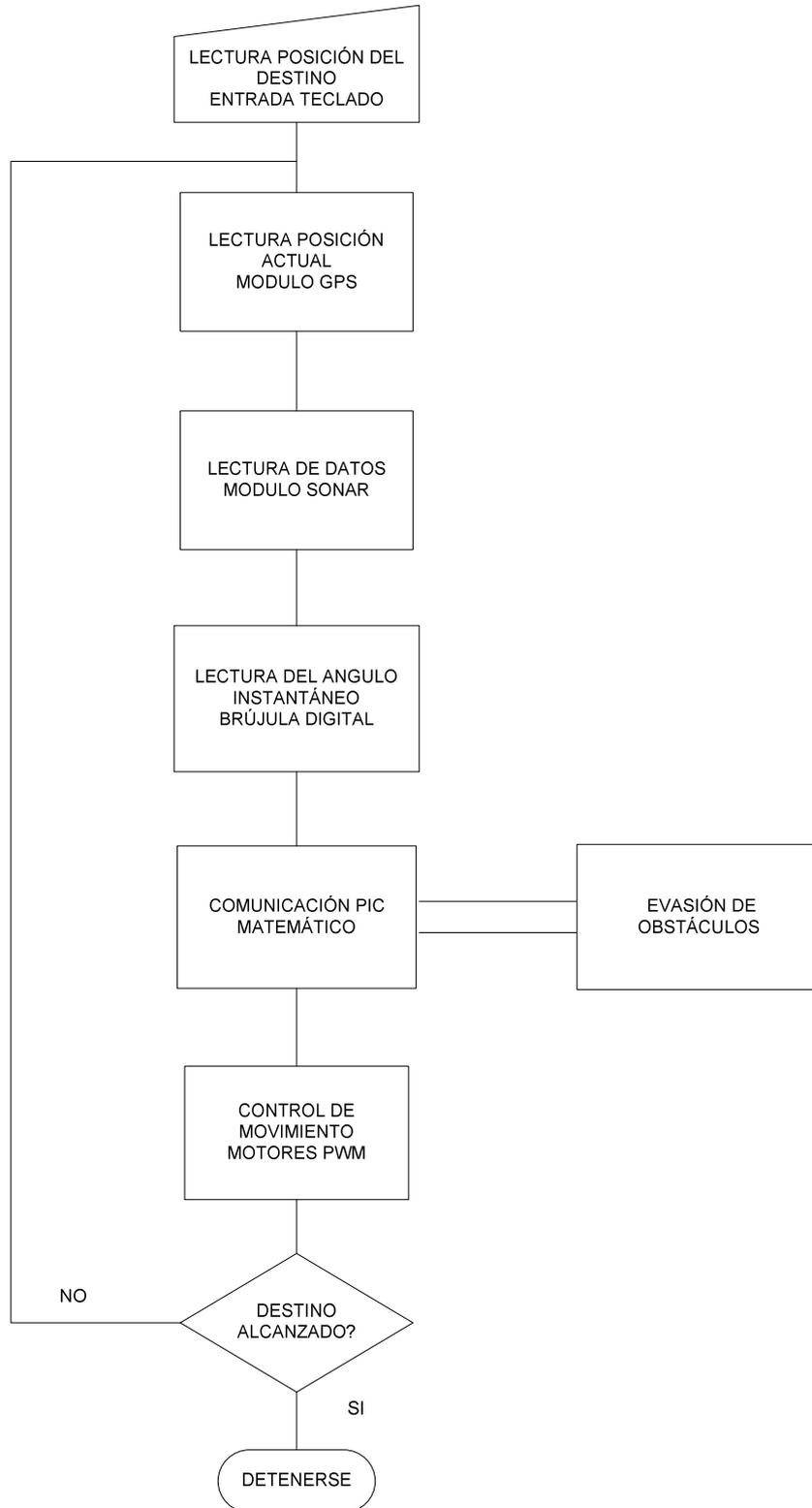


Figura IV.12. Secuencia principal para la navegación GPS

#### **4.4.2. Comunicación con el PIC matemático**

Aprovechando el método de comunicación empleado por el módulo GPS y los sensores ultrasónicos, la comunicación entre el PIC16F877A de la tarjeta principal y el PIC matemático 18F452 se realizó de la misma manera.

El PIC 16F877A genera una trama de caracteres ASCII que comienza con los caracteres "PS" seguido de la latitud y longitud del punto de salida, después se genera otra trama que comienza con los caracteres "PD" seguido de la latitud y longitud del punto de destino a alcanzar. Por último se envían las lecturas de los cuatro sensores ultrasónicos utilizados y de la brújula digital.

El PIC 18F452 está constantemente comprobando su canal de comunicación a la espera de los datos que le envía el PIC 16F877A, cuando los datos son recibidos, realiza todos los cálculos necesarios y devuelve la respuesta de dicho cálculo en formato ASCII que comienza con los caracteres "RM" seguidos de cuatro caracteres que representan el ángulo de dirección que el robot debe seguir, a continuación dos caracteres que indica la dirección de movimiento y por último tres caracteres que se asocian a la velocidad de giro de los motores. El diagrama para la comunicación entre ambos PICs s mostrado en la figura IV.13.

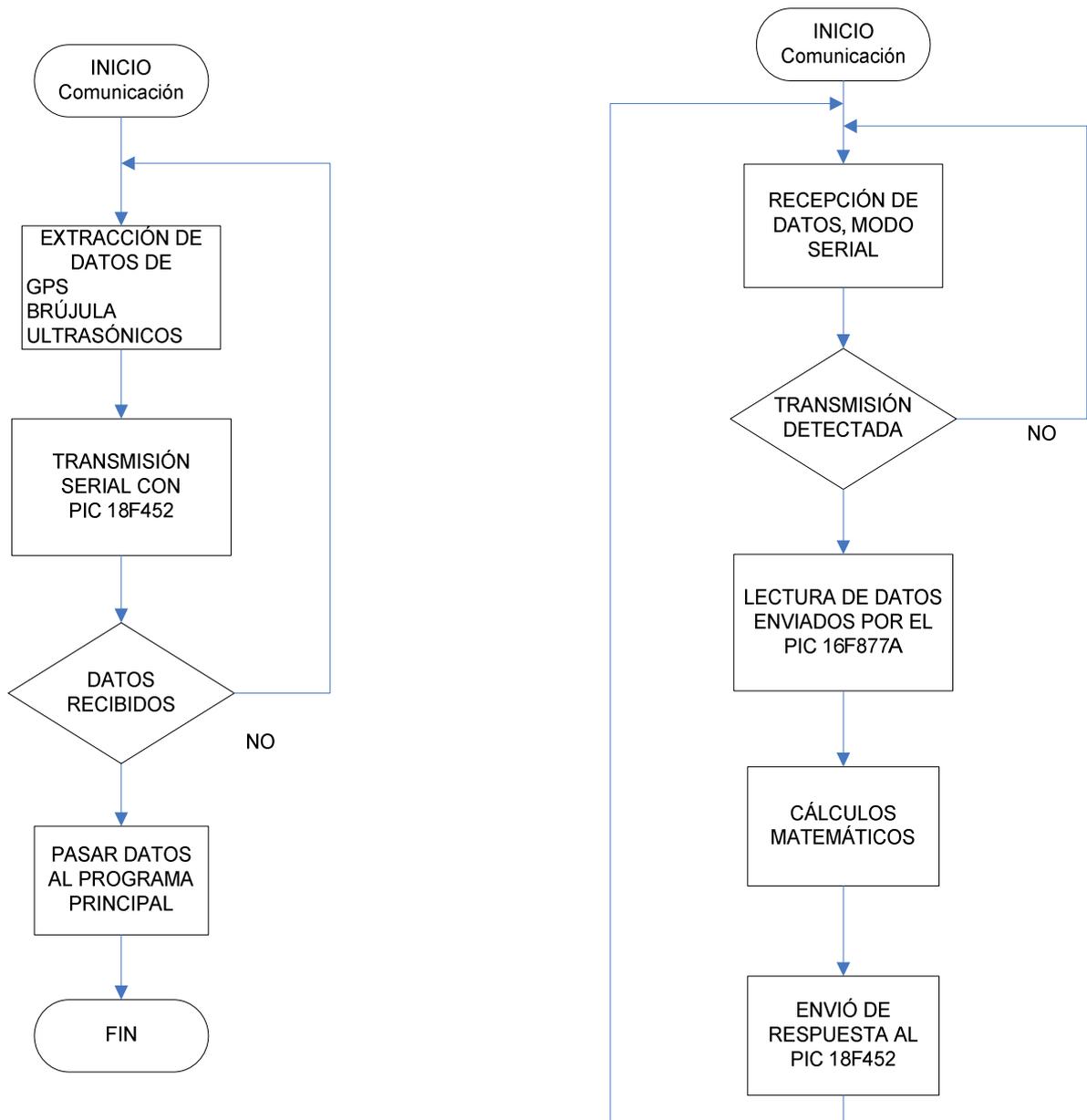


Figura IV.13: Algoritmos de comunicación para el PIC 16F877A y 18F452

#### 4.4.3. Programa de evasión de obstáculos

El programa creado para el cálculo matemático sobre un PIC 18F452 fue editado bajo el entorno de programación MikroBasic de MikroElectronica, este software de programación cuenta con una gran cantidad de librerías útiles que facilitan la edición de programas para los PICs de Microchip. La librería matemática y la capacidad de realizar cálculos con datos decimales fueron las principales características para la elección de este software.

## **El método del campo de fuerzas virtuales (VFF)**

Para la evasión de obstáculos se implementó la teoría del Campo de Fuerzas Virtuales descrita por Borenstein J. (3), con esta técnica el robot tiene que viajar hacia un objetivo, que ejerce una fuerza atractiva sobre él, simulando un tirón gravitatorio, en nuestro caso el punto determinado por las coordenadas de latitud y longitud.

Esta fuerza es mayor, cuanto más lejos esté el robot del objetivo, la dirección de la fuerza de atracción está determinada por el rumbo. Sin embargo no es esta la única fuerza que actúa sobre el robot. Puesto que el fin último de esta técnica es evitar los obstáculos más cercanos al móvil, estos generan una fuerza repulsiva sobre el robot y es mayor cuanto más cerca está de éste.

La atracción y repulsión son modeladas como vectores que representan fuerzas virtuales a las que se ve sometido el robot y poseen una magnitud y un ángulo relativo al robot mismo. Así el movimiento del robot se ve gobernado por la suma vectorial de todas las fuerzas que afectan al robot. Esta técnica garantiza el alejarse de los obstáculos y recuperar el rumbo hacia el objetivo una vez se han superado éstos.

Al hacer un barrido de mediciones de distancia con los sensores de ultrasonido es posible encontrar obstáculos en algunas posiciones o no encontrar ninguno en absoluto, caso en el cual el robot no experimenta repulsión. En el caso de encontrar un obstáculo hacia un determinado ángulo, el robot "siente" una fuerza en la misma dirección pero en sentido contrario, cuya magnitud depende de la distancia medida. En caso de encontrar varios obstáculos al hacer el barrido, todas las fuerzas se suman vectorialmente para obtener un vector repulsivo resultante (Figura IV.14).

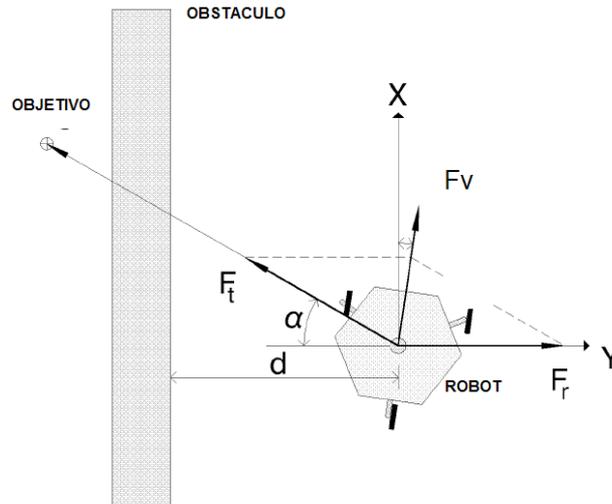


Figura IV.14: Campo de fuerzas repulsivas generadas por la presencia de obstáculos en el área de trabajo del robot, Borenstein J. (3).

Como sólo se dispone del ángulo relativo al robot pero no se conoce a qué distancia se encuentra la meta, se da a esta fuerza un valor constante en magnitud. Entonces, durante el movimiento del robot, se computa continuamente la sumatoria vectorial que permite obtener la dirección deseada de movimiento para evitar los obstáculos, teniendo en cuenta en todo momento la meta a alcanzar.

#### 4.5. Calculo del rumbo (Vector Fuerza Atracción)

El rumbo es el ángulo formado por la línea que une el punto de partida y el destino con respecto al polo norte como es mostrado en la figura IV.15, el robot debe seguir este ángulo para alcanzar un objetivo específico, dadas sus coordenadas geográficas (latitud y longitud). El tipo de navegación utilizado es la navegación loxodrómica.

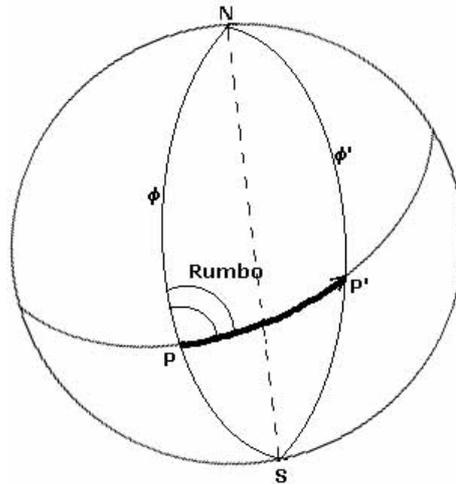


Figura IV.15: Determinación del rumbo, China (6).

Se denomina loxodrómica, a la línea que une dos puntos cualesquiera de la superficie terrestre cortando a todos los meridianos con el mismo ángulo, SOUTHWINGS (27). La loxodrómica, por tanto, es fácil de seguir manteniendo el mismo rumbo. En el proceso de cálculo se asumen distancias menores a 600 millas náuticas y latitudes menores de  $60^\circ$ , utilizando fórmulas que consideran diferencia de latitud, apartamiento, diferencia de longitud, que se basa en la resolución de un triángulo rectángulo a través del uso de trigonometría plana (figura IV.16).

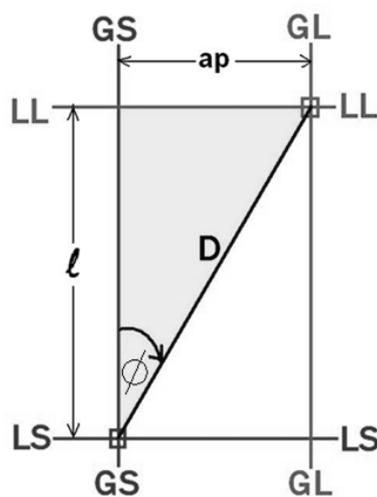


Figura IV.16: Triángulo formado por el punto de salida, llegada y el polo norte, SOUTHWINGS (28).

El cálculo del rumbo es realizado mediante las siguientes formulas

$$\ell = D * \cos RV \quad (4.1)$$

$$ap = D * \sen RV \quad (4.2)$$

$$LL = LS \pm \ell \quad (4.3)$$

Latitud media en el mismo hemisferio

$$Lm = \frac{L1 + L2}{2} \quad (4.4)$$

Latitud media en distinto hemisferio

$$Lm = \frac{L1 - L2}{2} \quad (4.5)$$

$$g = ap \div \cos Lm \quad (4.6)$$

$$GL = GS \pm g \quad (4.7)$$

$$\emptyset = \arctg(ap \div \ell) \quad (4.8)$$

Donde:

**PS** = Punto de salida

**PL** = Punto de Llegada

**LS** = Latitud de Salida

**GS** = Longitud de Salida

**LL** = Latitud de Llegada

**GL** = Longitud de Llegada

**D** = Distancia navegada en Millas Náuticas.

**RV** = Rumbo verdadero en grados

$\ell$  = Diferencia de Latitud; representa la diferencia, en minutos o grados de latitud, hacia el N o hacia el S, entre el PS y el PL.

Signos diferentes entre LS y "l", se restan y se conserva el del mayor para la LL.

Signos iguales entre LS y "l", se suman y se conserva el del mayor para la LL.

**Ap** = Apartamiento; representa la distancia en MN, hacia el E o W, entre el PS y PL.

Se usa solo en millas náuticas, independiente de su magnitud y nunca se convierte a grados de Longitud. Su signo E o W es función del RV.

**Lm** = Latitud media; es el promedio entre LS y LL

**g** = Diferencia de Longitud; representa la diferencia, en minutos o grados de longitud, entre el PS y el PL. Adopta el mismo signo del ap.

Signos diferentes entre GS y g, se restan y se conserva el del mayor para la GL.

Signos iguales entre GS y "g", se suman y se conserva el del mayor para la GL.

El proceso de cálculo es realizado en el PIC 18F452 (figura IV.17) a través de la función "**rumbo**" Anexo 2, el valor obtenido determina el ángulo que el robot debe seguir por medio de la brújula digital para alcanzar el destino, además este ángulo determina la dirección del vector fuerza de atracción necesaria para el algoritmo VFF.

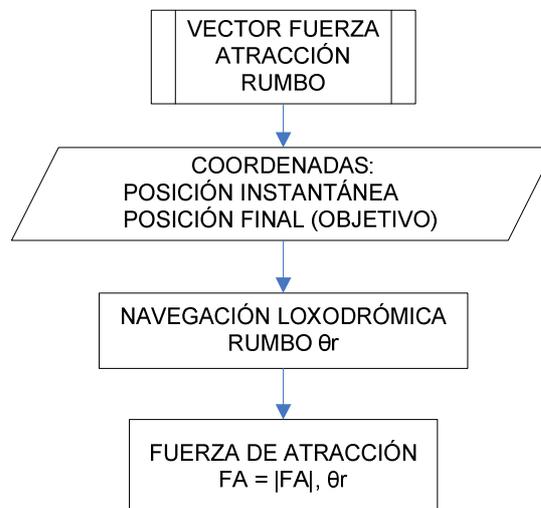


Figura IV.17: Algoritmo para el cálculo del vector fuerza de atracción

### 4.5.1. Calculo del vector fuerza repulsiva

El primer aspecto a tener en cuenta al hacer la implementación se refiere a la relación que debe existir entre la distancia medida por los sensores ultrasónicos y la magnitud de la fuerza repulsiva que experimentaría el robot debido a ella.

La fuerza repulsiva es inversamente proporcional a la distancia a la que se haya detectado el obstáculo dada por la ecuación 4.9 , llegando a ser de valor cero en magnitud para una distancia determinada, que se fijo a 120 pulgadas, aproximadamente 3m.

$$Fr = \frac{k}{d} \quad (4.9)$$

Donde

k=constante de proporcionalidad

d= distancia a un obstáculo

La relación utilizada para el cálculo de la fuerza repulsiva en los trabajos anteriores realizados por Borenstein (3), es que la fuerza repulsiva es inversamente proporcional al cuadrado de la distancia al obstáculo ecuación (4.10). Esta relación asegura que la fuerza repulsiva sea mayor cuando el obstáculo se encuentra muy cerca del robot.

$$Fr = \frac{k}{d^2} \quad (4.10)$$

Con la fórmula m el modulo del vector fuerza repulsiva es calculado para cada sensor ultrasónico, cada sensor ultrasónico está ubicado a un ángulo específico con respecto al eje X, este ángulo determina la dirección de cada vector fuerza lo que permite calcular las componentes X,Y. La sumatoria de todas estas fuerzas repulsivas ecuación (4.11) determina la fuerza repulsiva total.

$$\sum_{1}^{4} Fr_x = FrT_x, \quad \sum_{1}^{4} Fr_y = FrT_y \quad (4.11)$$

La función “**fuerza**” implementada para el PIC18F452 contiene todas las instrucciones necesarias para determinar el vector fuerza repulsiva total (figura IV.18). Antes de calcular las componentes X, Y de cada vector fuerza, se realiza un desplazamiento del ángulo de dirección de cada vector fuerza basado en el ángulo instantáneo en la dirección de desplazamiento del robot.

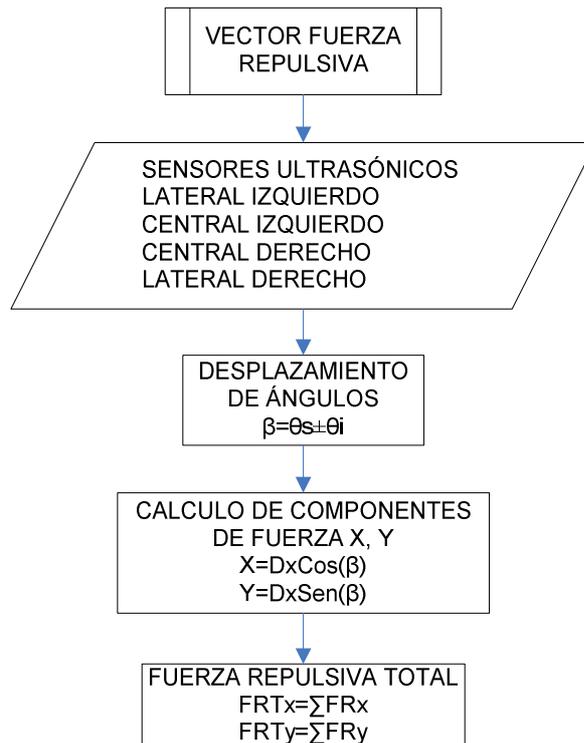


Figura IV.18: Algoritmo para el cálculo del vector fuerza repulsiva

#### 4.5.2. Calculo del vector fuerza resultante

El robot siempre tenderá a avanzar hacia el frente mientras el vector resultante (de la sumatoria vectorial entre Atractivo y Repulsivo) se encuentre en dirección hacia delante y gira en el sentido correspondiente cuando el vector apunta en otra dirección.

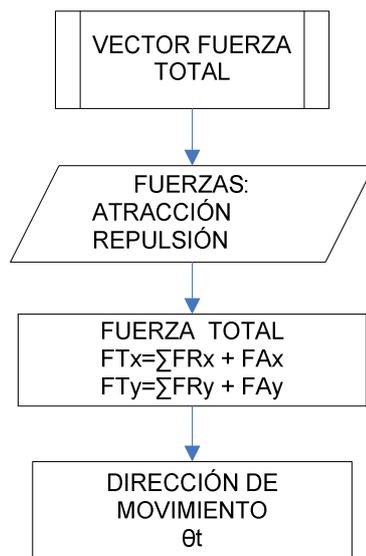


Figura IV.19: Algoritmo para la determinación de la dirección del movimiento

Dado que no se dispone de ninguna herramienta de retroalimentación del movimiento (odometría) no es posible ordenar al robot realizar un giro de un ángulo determinado, por lo tanto es necesario hacer un muestreo continuo del ángulo del vector atractivo y una lectura del ángulo proporcionado por la brújula digital, para mantener una buena relación de giro hacia el objetivo deseado.

Es importante notar que las magnitudes de los vectores atractivo y repulsivo afectan directamente el cálculo de la sumatoria ya que el vector resultante tendrá siempre un ángulo menor con respecto a la componente que tenga mayor magnitud. Es decir, para dos vectores que conforman un determinado ángulo, si el atractivo es mayor en magnitud que el repulsivo, el vector resultante describirá un ángulo agudo con respecto al primero, resultando al contrario cuando el vector repulsivo es mayor. Para la decisión de evasión sólo se tiene en cuenta el ángulo del vector resultante, que gracias a lo anterior permite reaccionar correctamente cuando existe un peligro latente por la presencia de obstáculos cercanos.

### 4.5.3. Control de movimiento por PWM

El algoritmo de evasión de obstáculos determina el ángulo con el cual el robot debe moverse, cuando no se detecta ningún obstáculo la navegación es realizada calculando el rumbo.

Para el control de movimiento PWM actúan tres variables, las cuales le indican al robot la dirección de movimiento, la velocidad a la que debe moverse y la relación de velocidad entre motores, para realizar un movimiento circular, todo esto permite que el robot esquive obstáculos a la vez que avanza hacia el objetivo.

La primera variable a considerar esta denominada *Direccion*, puede toma un valor de 0/1, un valor de 0 indica un movimiento hacia la derecha, mientras que un 1 indica un movimiento hacia la izquierda. Este valor está determinado por la diferencia entre el ángulo instantáneo  $\theta_i$  que describe el robot y el ángulo determinado por el algoritmo de evasión de obstáculos  $\theta_t$ , ecuación ( 4.12).

$$\Delta\theta = \theta_i - \theta_t \quad \text{dir} \begin{cases} 1, & \Delta\theta > 0 \\ 0, & \Delta\theta < 0 \end{cases} \quad (4.12)$$

La segunda variable se denomina *Motion*, puede tomar un valor de entre 1 y 5, y determina el tipo de movimiento que se va a realizar, que va desde un movimiento completamente recto con un valor de 1, hasta un giro completo en el mismo eje cuando toma el valor de 5, esto se logra variando la relación de velocidad que existe entre el motor derecho e izquierdo. Este valor está relacionado con la ecuación 4.13 que no es nada más que la misma ecuación 4.12 con algunas unas variaciones.

$$\Delta\theta = \theta_i - \theta_t \text{ Motion} \begin{cases} 1, & 0 \leq |\Delta\theta| < 5 \\ 2, & 5 \leq |\Delta\theta| < 35 \\ 3, & 35 \leq |\Delta\theta| < 60 \\ 4, & 60 \leq |\Delta\theta| < 85 \\ 5, & 85 \leq |\Delta\theta| < 180 \end{cases} \quad (4.13)$$

Por último el valor *Velocidad* determina la relación de porcentaje de la onda PWM del programa principal y está en función a la distancia del objeto más próximo, haciendo una subdivisión que se muestra en la ecuación (4.14)

$$Velocidad = K \begin{cases} 0, & 6 \leq |Distancia Mn| < 30 \\ 70, & 31 \leq |Distancia Mn| < 50 \\ 90, & 51 \leq |Distancia Mn| < 80 \\ 100, & 81 \leq |Distancia Mn| < 150 \\ 120, & 150 \leq |Distancia Mn| < 250 \end{cases} \quad (4.14)$$

Donde: Distancia Mn representa la menor distancia detectada por cualquiera de los sensores, estos valores controlan la velocidad máxima a la que puede moverse el robot, la misma que está determinada por el control PWM que acepta valores de entre 0 y 250, donde 0 detiene completamente el motor y el 250 permite la velocidad máxima que puede alcanzar el motor, k es una constante que determina el valor de velocidad deseado con respecto a la distancia mínima detectada.

De donde se pudo analizar que si bien el movimiento a mayor velocidad reduce el tiempo para alcanzar el objetivo, también aumenta las probabilidades de tener una colisión, así mismo un valor demasiado bajo ocasiona que los motores no puedan romper la inercia y por ende puedan llegar a dañar el circuito de control de potencia.

Por estas razones se escogió trabajar con la ecuación (4.14) que presenta una buena relación entre la distancia próxima a un obstáculo y la velocidad de respuesta, como se puede ver en la siguiente figura.

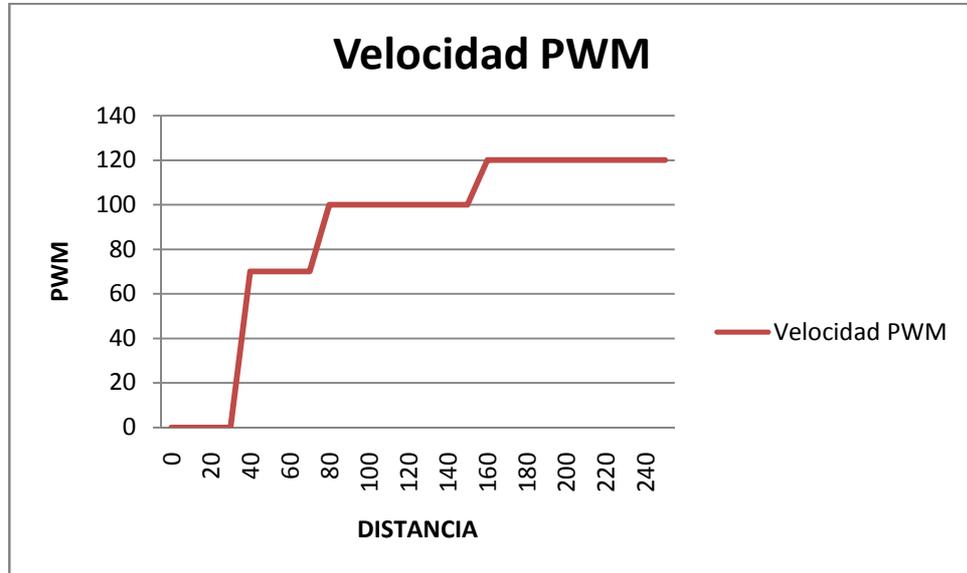


Figura IV.20. Curvas de respuesta entre el ángulo  $\theta_w$  y el control de velocidad PWM

Con los valores de estas tres variables el algoritmo de evasión de obstáculos entra en acción y cambia la dirección del movimiento hasta que el obstáculo haya sido evadido, después retoma el rumbo y continúa en línea recta.

Este ciclo de lectura de los posibles obstáculos a través del módulo sonar, el ángulo de movimiento instantáneo de la brújula digital, la posición geográfica del módulo GPS y el cálculo del algoritmo de evasión de obstáculos VFF es continua hasta que la comparación de las coordenadas del objetivo y de la posición actual sean aproximadamente las mismas, esto debido a la precisión del módulo GPS. De ser encontrado el objetivo el robot se detiene y espera el ingreso de un nuevo objetivo o destino.

## 4.6. Implementación

La implementación del software de control para el robot se realizó en Microcode y Microbasic. Todo el código fuente completo en formato para compilar se puede encontrar en el anexo C y D a continuación se detalla parte de las principales funciones utilizadas para la implementación del sistema, aclarando que no es el programa completo sino una guía para el entendimiento del sistema.

### 4.6.1. Comunicación con el módulo receptor GPS

La transmisión de los mensajes se realizó por un canal de comunicación serial bajo los parámetros mostrados en la tabla IV.5, se utilizó la instrucción SERIN2 de Microcode Studio:

PARÁMETRO	ESPECIFICACIÓN
Interface	Simple UART
Bit rate	4800 bps
Start bit	1 bit
Stop bit	1 bit
Data bit	8 bit
Paridad	None
Transmisión datos	NMEA 0183 Ver3.01

Tabla IV.5: Parámetros de comunicación con el módulo receptor GPS

La subrutina *EXTRAC\_COOR\_GPS* contiene el juego de instrucciones que permiten la extracción de las coordenadas del punto en donde se encuentra ubicado el robot. La instrucción SERIN2 es utilizada para extraer el mensaje GGA de la siguiente forma:

*Serin2*

```
GPSin,baudGPS,2000,GPS_TIMEOUT,[wait("$GPGGA,"),GPS_RX[0],...,GPS_RX[n]]
```

Donde:

**GPSIn** corresponde a la entrada de datos para la comunicación serial que es el pin PORTD.0 del PIC16F877A.

**BaudGPS** es el valor que representa la configuración de los parámetros de comunicación serial mostrados en la tabla IV.5, para 4800 baudios, este valor corresponde a 188 que es calculado mediante la siguiente fórmula:

$$\text{Modo} = (1000000 / \text{baudios}) - 20$$

**2000** es el tiempo máximo de espera sin recibir comunicación desde el transmisor, de sobrepasar este tiempo se ejecutan las instrucciones dentro de *GPS\_TIMEOUT*.

Mediante la instrucción *wait("\$GPGGA,")*, se espera que la secuencia "\$GPGGA," sea recibida, antes de empezar a almacenar en el vector *GPS\_RX[n]* los caracteres ASCII de cada parámetro del mensaje GGA del estándar NMEA.

#### 4.6.2. Comunicación con el módulo sonar

El grupo de instrucciones que contralan la comunicación y el proceso de datos de los sensores ultrasónicos se encuentran agrupados dentro de la subrutina *MAX\_MAIN*. La lectura se realizó igual que se lo hizo con el modulo GPS, con la instrucción *SERIN2* de Microcode,

La instrucción para la comunicación serial con el modulo sonar es la siguiente:

**Serin2**

***MAXSin,baudMAX,100,MAX\_TIMEOUT,[wait("R"),GPS\_RX[0],GPS\_RX[1],GPS\_RX[2]]***

Donde:

**MAXSin** corresponde a la entrada de datos para la comunicación serial que es realizada por el pin PORTA.4 del PIC16F877A, esta entrada es configurada por software.

**baudMAX** es el valor que representa la configuración de los parámetros de comunicación serial mostrados en la tabla IV.5, para 9600 baudios, este valor corresponde a 16468.

**100** es el tiempo máximo de espera sin recibir comunicación desde el transmisor, de sobrepasar este tiempo se ejecutan las instrucciones dentro de *MAX\_TIMEOUT*.

Mediante la instrucción *wait("R")*, se espera que el carácter "R" sea recibido, antes de empezar a almacenar en las variables MAX\_LI, MAX\_LD, MAX\_CI, MAX\_CD que representan los sensores ubicados en extremo lateral izquierdo, lateral derecho, centro izquierdo y centro derecho respectivamente.

#### 4.6.3. Comunicación con la brújula electrónica

Las instrucciones de comunicación con la brújula electrónica se encuentran dentro la subrutina *CMP03*. La instrucción *I2CREAD* de Microcode es usada para leer los datos de la brújula de la siguiente forma.

```
I2CREAD dpin,cpin,%11000000,2,[bearing.Highbyte,bearing.lowbyte]
```

Donde

***Dpin*** representa el pin de datos PORTC.4 del PIC 16F877, SDA del estándar I2C.

***Cpin*** representa el pin de reloj PORTC.4 del PIC 16F877, SCL del estándar I2C.

***%11000000*** es la instrucción de control que le indica a la brújula digital que se realizará una lectura de datos.

***2*** es la dirección del registro leído de la brújula digital.

La variable **bearing** almacena el valor del ángulo formado en la posición actual con respecto al norte magnético.

No se requiere proceso adicional alguno ya que el valor entregado por la brújula esta en formato binario fácil de comprender por el PIC.

#### 4.6.4. Comunicación con el PIC matemático

El juego de instrucciones para la comunicación con el PIC matemático se encuentra dentro de la subrutina TX\_RX\_18F, básicamente es una comunicación serial a 9600 baudios de dos vías, del un extremo se envían todas las variables necesarias para el calculo del rumbo y para evitar obstáculos, del otro lado se recibe los valores de RUMBO, DIRECCION y VELOCIDAD en las variables del mismo nombre. Como en todo el sistema la instrucción serout2 y serin2 son utilizadas para lograr la comunicación serial, las instrucciones más importantes se detallan a continuación.

```
serout2 P18out,baudP18,["PS",GP_LATS[2],...,GP_LATS[10],"OK"]
serout2 P18out,baudP18,[GP_LONS[0],GP_LONS[1],...,GP_LONS[9],"OK"]
serout2 P18out,baudP18,["PD", GP_LATD[2],..., GP_LATD[10],"OK"]
serout2 P18out,baudP18,[ GP_LOND[0],...,GP_LOND[9], "OK"]
serout2 P18out,baudP18,[dec MAX_Li,"OK"]
serout2 P18out,baudP18,[dec MAX_Ci,"OK"]
serout2 P18out,baudP18,[dec MAX_Cd,"OK"]
serout2 P18out,baudP18,[dec MAX_Ld,"OK"]
serout2 P18out,baudP18,[BEARING,"OK"]
serin2 P18in,baudP18,PIC_Timeout,[wait("RM"),GPS_RX[0],...,GPS_RX[8]]
```

Donde:

**P18out** es el pin PORTC.6 del PIC 16F877A que se encarga de la transmisión de los datos.

**baudP18** es la velocidad de transmisión de la comunicación serial, para 9600 baudios toma el valor de 84.

**GP\_LATS, GP\_LONS, GP\_LATD, GP\_LOND** corresponden a los vectores que almacenan las coordenadas del punto inicial y del objetivo LAT corresponde a la latitud, LON corresponde a la longitud mientras que S y D corresponden al punto de Salida y al punto de destino respectivamente.

Las variables **MAX\_LI, MAX\_LD, MAX\_CI, MAX\_CD** corresponden a los valores obtenidos de la comunicación con el modulo sonar y la variable **bearing** corresponde al valor obtenido de la lectura de la brújula electrónica.

Mediante la instrucción *wait("RM")*, se espera a que los caracteres "RM" sean recibidos, después se almacenan los datos en el vector GPS\_RX para la posterior extracción del rumbo, dirección y velocidad.

#### **4.6.5. Programa de evasión de obstáculos**

El código para la evasión de obstáculos fue realizado en el entorno de programación MikroBasic debido a que este cuenta con las librerías necesarias para realizar cálculos matemáticos, además puede compilar el código para el PICs de gama alta 18F452 que se utilizó para este proyecto.

Las formulas de cálculo del rumbo y de evasión de obstáculos por el método de fuerzas virtuales fueron implementadas como se explico en la etapa de diseño, acoplándolas al lenguaje de programación que fue utilizado.

El código completo se encuentra en el archivo "VFF.pbas" del **Anexo M**, en el cual se puede encontrar todas las funciones y formulas explicadas en la etapa de diseño, las principales funciones son:

**latlong(LA, LG, dirNS, EW, rx, SD)**, Extrae las coordenadas del punto geográfico y las convierte en un ángulo en radianes para su posterior utilización.

***rumbo(Lts1, Los1, ltd1, lod1, La\_s, Lg\_s, La\_d, Lg\_d)***, Realiza el cálculo del rumbo, este valor es el ángulo en radianes que el robot debe seguir para llegar al destino.

***Frepulsiva(Qr, ModFr, Frx, Fry)***, De los valores obtenidos por el modulo sonar, se calcula el vector fuerza repulsiva.

***Fuerza\_virtual()***, Es la función más importante, es la implementación misma del algoritmo de fuerzas virtuales, como resultado se obtiene el ángulo que el robot debe seguir para no chocar con los obstáculos que las limitaciones del modulo sonar puede alcanzar.

#### 4.7. Pruebas y Validación

Las principales pruebas se realizaron en el simulador hasta llegar a constituir el programa completo, después de realizar un incremento se paso a comprobar que el algoritmo funcione correctamente para luego añadir otro incremento, cada incremento está relacionado con el control y comunicación de un sensor y la construcción del algoritmo de evasión de obstáculos, cabe recalcar que las pruebas se basaron en la implementación correcta del sistema para después pasar a las pruebas físicas del robot completo, las pruebas físicas se explican en el siguiente capítulo.

INCREMENTOS	ACCION DESARROLLADA
Incremento 1	Control de display y teclado
Incremento 2	Comunicación Modulo GPS
Incremento 3	Comunicación Modulo sonar
Incremento 4	Comunicación brújula electrónica
Incremento 5	Control puente H con PWM
Incremento 6	Comunicación PIC 18F452 y PIC 16F877A
Incremento 7	Calculo del rumbo de navegación
Incremento 8	Algoritmo fuerzas virtuales
Incremento 9	Integración del programa principal
Incremento 10	Software de navegación final

Tabla IV.6: Modelo incremental (Incrementos desarrollados)

## **CAPITULO V**

### **ANÁLISIS DE PRUEBAS Y RESULTADOS**

A continuación se presentan las pruebas básicas que se realizaron a los diferentes módulos que conforman el robot móvil y los resultados obtenidos que ayudaran a la mejora del robot y la determinación importantes conclusiones sobre el proyecto.

#### **5.1. Prueba y análisis del Modulo Sonar**

##### **5.1.1. Prueba del los sensores ultrasónicos**

Las pruebas llevadas a cabo fueron realizadas para comprobar el correcto funcionamiento de cada sensor ultrasónico y poder determinar las características de los posibles materiales que sería capaz de detectar.

### **5.1.2. Diseño de la prueba**

La prueba se llevo a cabo en un área relativamente despejada al frente de cada sensor y con paredes de concreto a los lados.

Se consideraron tres muestras tomadas por dos sensores, para dos objetos con características completamente distintas ubicados a distancias de entre 0 y 120 pulgadas, que es la distancia de trabajo proyectada para el presente trabajo, con una variación de 24 pulgadas.

La variable a medir es el promedio de las tres muestras para poder compararlas con la distancia real al objeto.

### **5.1.3. Resultados de la prueba**

De las pruebas en distintos materiales se pudo determinar que los objetos con superficies demasiado lisas o irregulares, producen grandes variaciones en la medición de la distancia e incluso la perdida de las mismas, así mismo, por las características de las ondas ultrasónicas, el ángulo de incidencia debe ser normal a la superficie del obstáculo, de lo contrario la onda rebota fuera del rango de cobertura de detección del sensor. El resumen de la prueba se muestra en la Tabla V.1

De la tabla V.1 es claro que las mediciones realizadas por los dos sensores frontales del robot móvil para dos tipos de materiales con características sumamente distintas, las lecturas tomadas para el caso en que se intenta detectar personas demuestran el efecto que tiene la forma del objeto, el error producido es alto, dejando en claro que la detección de personas es una labor sumamente difícil para el sensor ultrasónico, algo muy distinto ocurre para el caso en el que se realizaron mediciones para una caja de cartón de 9 x 18 pulgadas ubicada al frente de los sensores con un ángulo de incidencia completamente perpendicular, en el que las lecturas son altamente confiables.

Tipo de material	Distancia	Sensor izquierdo				Sensor derecho			
	Pulgadas	M1	M2	M3	Promedio	M1	M2	M3	Promedio
PERSONAS	0-6	6	6	6	6	6	6	6	6
	24	51	25	26	34	52	71	117	80
	48	25	25	25	25	25	112	68	68
	72	213	48	246	169	209	127	216	184
	96	90	182	110	127	170	69	68	102
	120	213	148	213	191	192	193	255	213
CAJAS DE 9X18 PULGADAS	0-6	6	6	6	6	6	6	6	6
	24	23	20	23	22	20	24	24	23
	48	43	43	43	43	43	43	43	43
	72	68	68	68	68	68	72	72	71
	96	96	96	96	96	154	216	93	154
	120	119	119	119	119	115	115	115	115

Tabla V.1: Mediciones obtenidas para dos tipos de materiales

Después de las pruebas realizadas se determinó que los obstáculos debían tener las siguientes características: una altura mínima de 24 pulgadas y un ancho mínimo de 10 pulgadas, materiales con superficies parcialmente lisas, y de preferencia objetos rectangulares, dejando en claro que existen condiciones para que puedan ser detectados, la principal es el ángulo de incidencia con el que los objetos se encuentren respecto al robot. Así mismo la distancia mínima a la que el robot puede estar de obstáculo se fijó a 40 pulgadas para garantizar que no exista una colisión.

#### 5.1.4. Análisis de la ubicación de los sensores

Cada sensor ultrasónico tiene un ángulo de emisión de 30 grados, pero solo entrega el valor de la distancia obtenida a un objeto, pero no el ángulo al que se lo detectó, por lo que la probabilidad más alta de detección se encuentra en el centro del cono formado por su patrón de emisión, basándose en esta presunción, se colocaron los sensores a 15° y 45 ° en forma simétrica con respecto al eje x, figura V.1.

La configuración mostrada en la figura V.1, tenía un ángulo de cobertura total teórica de  $120^\circ$  para la detección de obstáculos, pero existían casos en los que los obstáculos no eran detectados, debido a que el espacio entre cada sensor deja un hueco del espacio en donde los sensores pueden actuar.

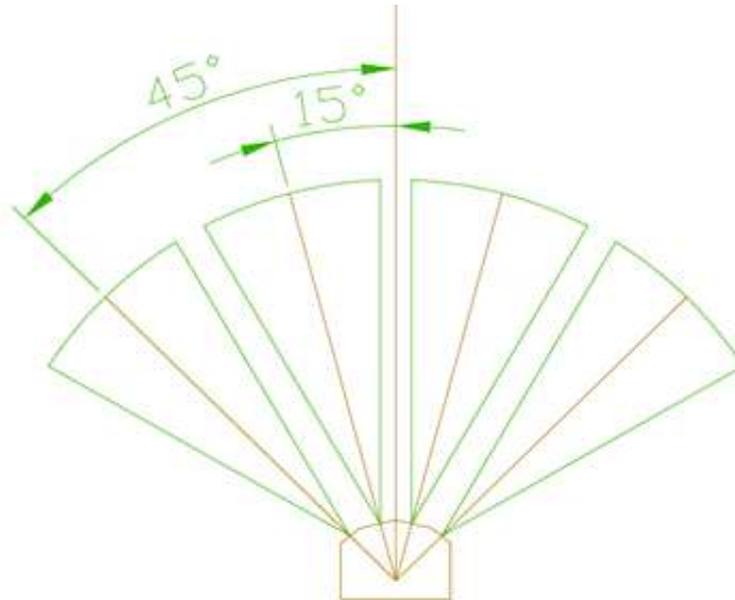


Figura V.1: Configuración inicial

Para evitar este problema se variaron los ángulos a los que están ubicados los sensores permitiendo una redundancia en el espacio de detección, los sensores fueron colocados a  $13^\circ$  y  $40^\circ$  en forma simétrica con respecto al eje x como se muestra en la figura V.2, esta variación conlleva a la reducción del ángulo de cobertura a  $109^\circ$ , este es un ángulo teórico, el ángulo de trabajo real es de  $80^\circ$  debido a que las lecturas a los extremos de los sensores no pueden ser tomadas como válidas ya que como se dijo antes la más alta probabilidad de detectar un obstáculo se encuentra en el centro del sensor, esto es  $40^\circ$ . De este análisis se pudo deducir que el robot sería incapaz de detectar obstáculos que se encuentren fuera del ángulo de cobertura, por lo tanto no se pueden esquivar objetos que se encuentren a los extremos y en la parte trasera del robot.

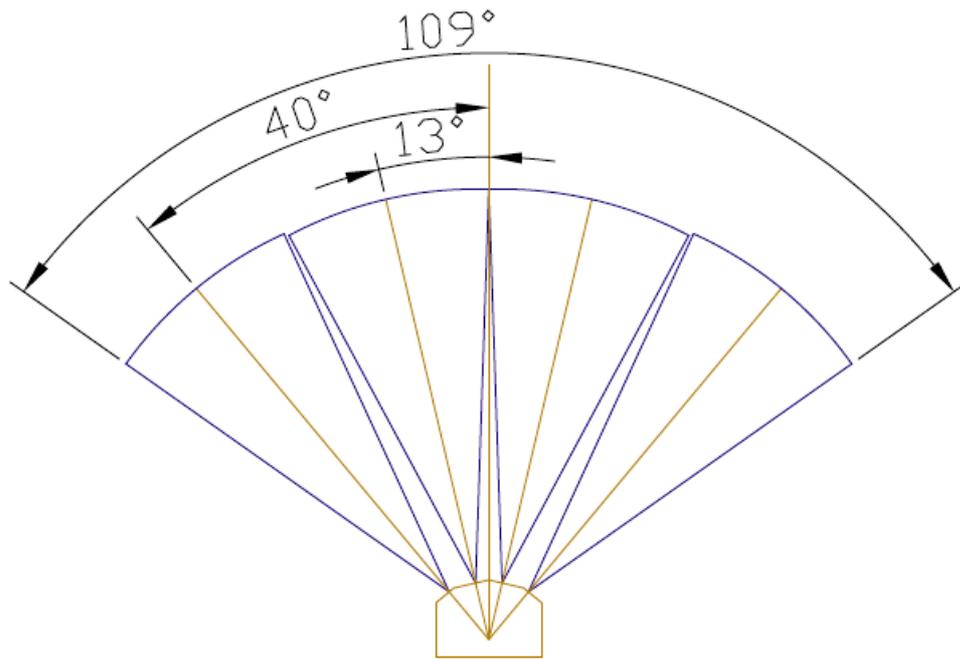


Figura V.2: Configuración final de los sensores ultrasónicos

## 5.2. Pruebas del algoritmo de Navegación

### 5.2.1. Prueba del algoritmo de cálculo del rumbo

La principal tarea que debe ejecutar el robot móvil es la de llegar hasta un objetivo, para lograrlo el robot lee constantemente los datos de posición, coordenadas geométricas, que entrega el modulo receptor GPS, pero, para poder hacerlo debe realizar el cálculo del rumbo. Si el cálculo del rumbo falla el robot nunca llegara al objetivo.

### 5.2.2. Diseño de la prueba

Las pruebas se llevaron a cabo con la ayuda del software Navigator Version 2.0.002, desarrollado por Peter A. Jacobson, especialmente para realizar cálculos de navegación marítima. La herramienta que se utilizó es la calculadora del círculo máximo mostrada en la figura V.3, ya que permite calcular el rumbo inicial para llegar de un punto a otro en la tierra.

El problema con Navigator fue su poca precisión decimal, fue desarrollado para navegar grandes distancias donde las variaciones entre coordenada inicial y final están dadas en los grados. Una limitación del robot móvil, es que no puede recorrer grandes distancias y las variaciones en las coordenadas, entre el punto de destino y el punto instantáneo, están dadas por los segundos. Por ello el cálculo del rumbo exige una mayor precisión decimal, pero fue de mucha ayuda para la prueba y corrección de nuestro algoritmo.

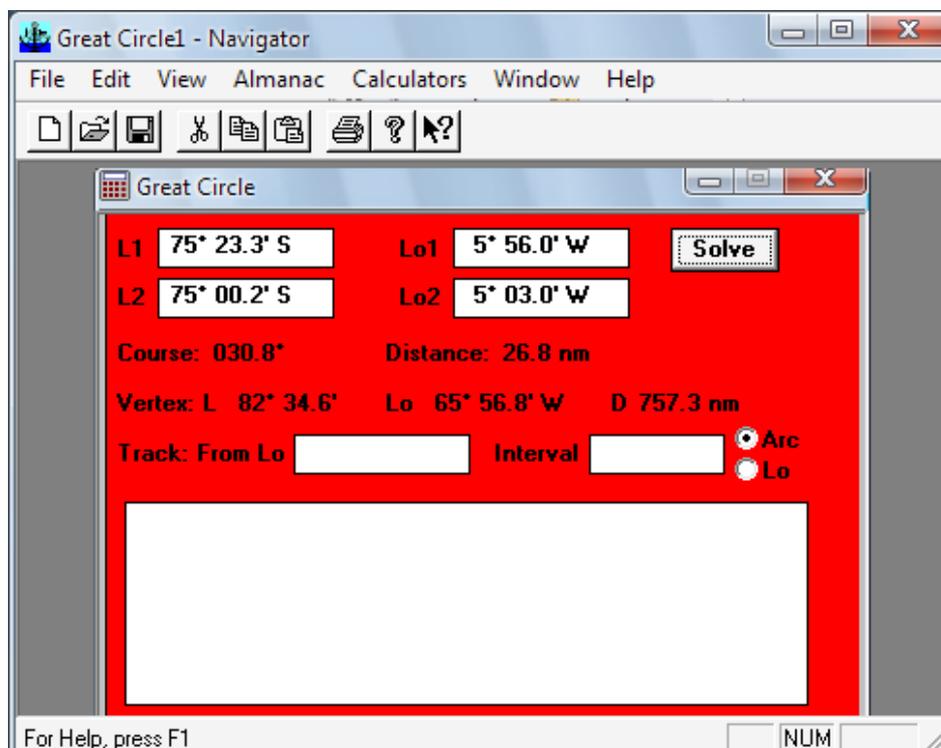


Figura V.3: Navigator, calculadora del círculo máximo

Se considero una posible trayectoria desde un punto de coordenadas  $1^{\circ}39'7586''$  S,  $78^{\circ}40'8681''$  W a otro de coordenadas  $1^{\circ}39'6348''$  S,  $78^{\circ}40'8334''$  W con una variación entre puntos de 14 8 segundos.

El parámetro medido fue la diferencia entre el ángulo arrojado por el algoritmo cargado en el PIC 18F452 "vff.pas" (asumiendo que no existen obstáculos) y el obtenido por medio del software Navigator.

### 5.2.3. Resultados de la prueba

Los resultados de la prueba están resumidos en la tabla V.2 mostrada a continuación. En donde se puede comprobar la gran precisión del algoritmo de cálculo de rumbo, la mayor variación se obtiene para el penúltimo dato donde se puede ver que las coordenadas de la posición de salida y llegada tienen una variación de 0.01 segundos, variación que es despreciable dado que el GPS puede sufrir una variación de hasta  $\pm 0.1$  segundos para un mismo punto.

Latitud salida				Longitud salida				Latitud destino				Longitud destino				rv(calculo)	rv(simulación)	Diferencia
g	m	s	dir	g	m	s	dir	g	m	s	dir	g	m	s	dir	grados	grados	ABS
1	39	7586	S	78	40	8681	W	1	39	6348	S	78	40	8334	W	15,651	15,652	0,001
1	39	7438	S	78	40	8639	W	1	39	6348	S	78	40	8334	W	15,626	15,626	0,001
1	39	729	S	78	40	8597	W	1	39	6348	S	78	40	8334	W	15,593	15,594	0
1	39	6994	S	78	40	8513	W	1	39	6348	S	78	40	8334	W	15,481	15,482	0,001
1	39	6846	S	78	40	8471	W	1	39	6348	S	78	40	8334	W	15,375	15,375	0
1	39	66,98	S	78	40	8429	W	1	39	6348	S	78	40	8334	W	15,18	15,183	0,003
1	39	655	S	78	40	8387	W	1	39	6348	S	78	40	8334	W	14,696	14,703	0,007
1	39	6402	S	78	40	8345	W	1	39	6348	S	78	40	8334	W	11,509	11,522	0,013
1	39	6347	S	78	40	8335	W	1	39	6348	S	78	40	8334	W	135,012	133,552	1,46
1	39	70	S	78	40	8334	W	1	39	6348	S	78	40	8334	W	0	0	0

Tabla V.2: Calculo del rumbo de navegación

De la tabla V.2 puede concluirse que el algoritmo desarrollado demostró ser altamente confiable y efectivo para la determinación del rumbo, pudiendo así continuar con la realización del proyecto.

### 5.3. Prueba del algoritmo de control de giro y velocidad

El objetivo de esta prueba fue determinar las relaciones de velocidad que debían existir entre los motores izquierdo y derecho para, lograr girar y avanzar en un ángulo deseado e incluso poder avanzar en línea recta.

### 5.3.1. Diseño de la prueba

Las pruebas se realizaron en un espacio abierto, simulando los modos de giro 2 al 4 determinados por la variable motion dentro del programa “vff.pas”, como se explico en el capítulo IV; escogemos un modo de giro y lo hacemos trabajar a velocidades con valores de PWM de entre 70 y 120 y determinamos la relación de velocidad que debe existir en entre el motor derecho e izquierdo para lograr el ángulo deseado.

Las variables medidas serán el factor de relación, con respecto al modo de giro.

### 5.3.2. Resultado de la Prueba

Los resultados de las pruebas se encuentran resumidos en la siguiente tabla

PWM	GIRO		
	2	3	4
70	10	20	30
90	10	20	40
100	20	40	60
120	20	40	60
RELACION	15	30	47,5

Tabla V.3: Relación para los diferentes modos de giro

De los resultados obtenidos en la tabla V.3. Se determino que para tener la respuesta del ángulo de giro para el modo 2 la variable Relación de potencia PWM debía tener un valor de 15 entre el motor derecho e izquierdo, para el modo 3 la relación debía tener un valor de 30 y por ultimo para el modo 4 la relación debía tener un valor de 47.5 pero se opto por el valor de 50. Estas relaciones son aplicadas en el control PWM suministrando más potencia a un motor que al otro logrando girar con mayor o menor rapidez a la derecha o izquierda.

## 5.4. Prueba del algoritmo de navegación GPS sin obstáculos

Esta prueba se realizó con el objetivo de conocer el comportamiento del robot móvil en un entorno libre de obstáculos con el algoritmo de de control principal, que es el responsable de orientar el robot hasta el punto de destino.

### 5.4.1. Diseño de la prueba

La prueba se llevó a cabo en un espacio abierto sin presencia de obstáculos, con el robot móvil ubicado en un punto A de coordenadas  $1^{\circ} 39' 7665''$  S,  $78^{\circ} 40' 9312''$  W, con orientación contraria a la posición de destino ubicada en un punto B de coordenadas  $1^{\circ} 39' 7693''$  S,  $78^{\circ} 40' 9251''$  W, como se observa en la figura V.4. Se considero 10 ensayos de la prueba, a partir del sexto ensayo se invirtió los papeles de los puntos A y B como destino y origen respectivamente (figura V.5). Se determinó si se alcanzó el objetivo deseado y el tiempo que le tomo.

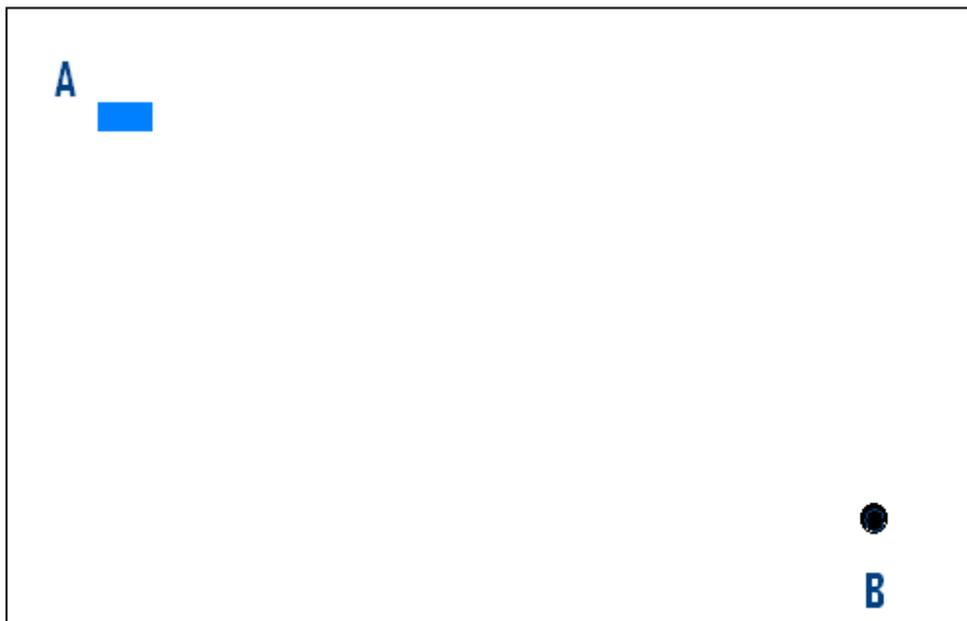


Figura V.4: Entorno de la prueba sin obstáculos, destino punto B.

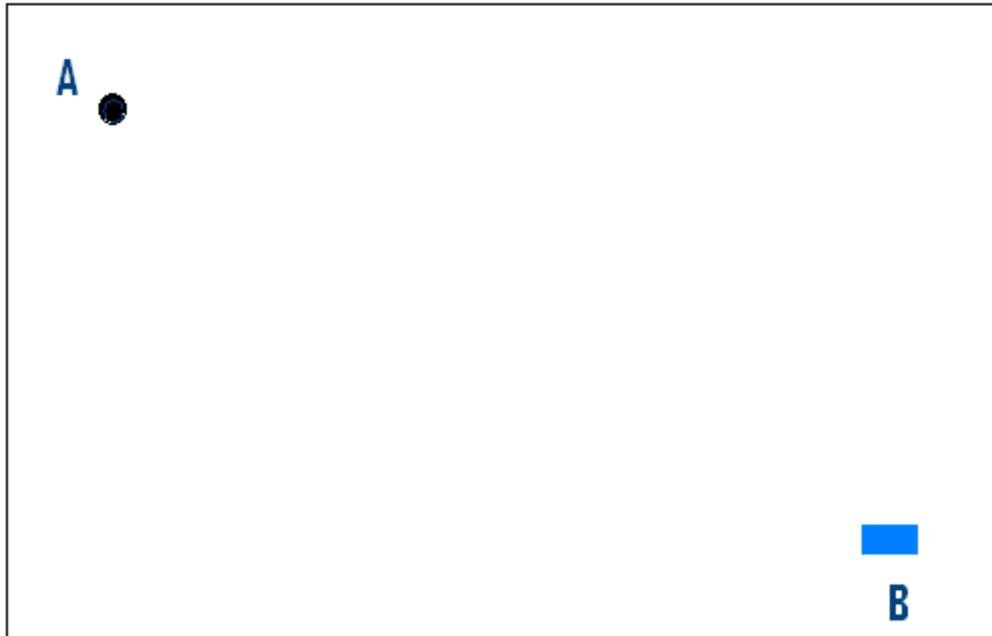


Figura V.5: Entorno de la prueba sin obstáculos, destino punto A

#### 5.4.2. Resultado de la prueba

En cada uno de los ensayos el robot alcanzo su objetivo, el tiempo que le tomo para alcanzar el objetivo en cada uno de los ensayos fue variable, esto se debió a la variación en las medidas tomadas por el modulo GPS ya que los valores de las coordenadas cambian para un mismo punto, además del desgaste de la batería que influye directamente en la velocidad del robot, lo que conlleva a la variación de la posición geográfica del destino en un máximo de 300 centímetros. Los resultados de la prueba se muestran en la tabla V.4. En promedio existe una distancia de 204 centímetros y un tiempo de 42 segundos al punto objetivo, durante la prueba al realizar el séptimo ensayo el robot no pudo alcanzar el objetivo, esto se debió a problemas de sincronización con el modulo GPS, este fue un evento aislado que no se ha repetido .

Ensayo	Destino alcanzado	Tiempo (s)	Error (cm)
1	si	26	140
2	si	25	260
3	si	35	300
4	si	32	200
5	si	36	210
6	si	33	100
7	no	120	0
8	si	34	300
9	si	36	130
10	si	47	200
PROMEDIO		42	204

Tabla V.4: Resultados prueba sin obstáculos

## 5.5. Prueba del algoritmo de navegación GPS con un obstáculo

Esta prueba se realizó con el objetivo de conocer el comportamiento del robot móvil en un entorno con un obstáculo entre el punto de origen y destino, con el algoritmo de de control principal que es el responsable de orientar el robot hasta el punto de destino y evitar las colisiones.

### 5.5.1. Diseño de la prueba

La prueba se llevó a cabo en un espacio abierto con un obstáculo en el centro del campo, con el robot móvil ubicado en un punto A de coordenadas  $1^{\circ} 39' 7665''$  S,  $78^{\circ} 40' 9312''$  W, con orientación contraria a la posición de destino ubicada en un punto B de coordenadas  $1^{\circ} 39' 7693''$  S,  $78^{\circ} 40' 9251''$  W, como se observa en la figura V.6 y V.7. Se considero 10 ensayos de la prueba y se determinó si se alcanzó el objetivo deseado y el tiempo que le tomo, además se determino si existieron colisiones, tomando en cuenta que se dice colisión cuando el objeto está en una posición que puede ser detectada por los sensores.

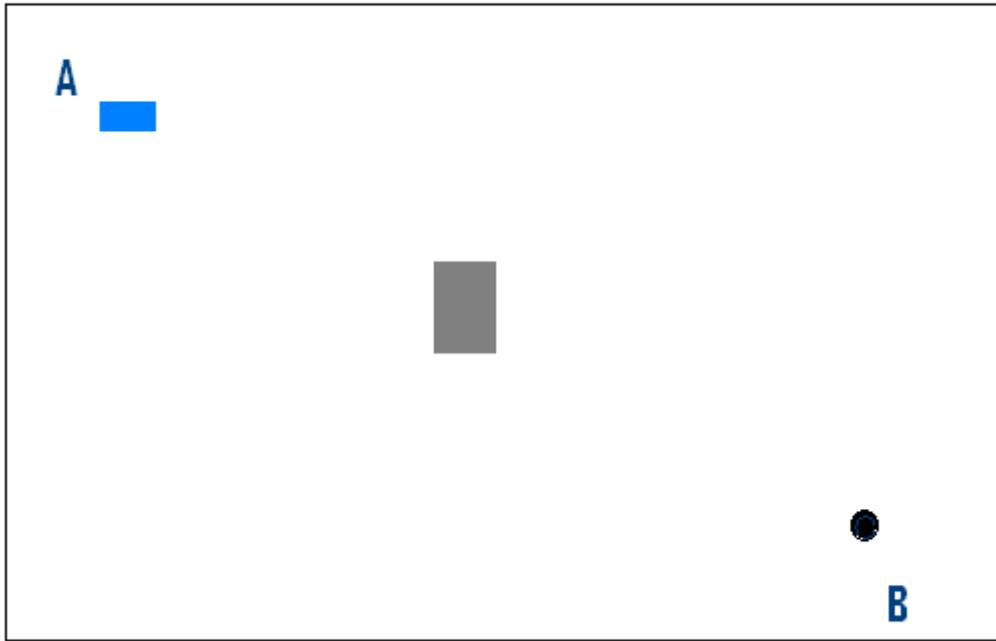


Figura V.6: Entorno de la prueba con un obstáculo, destino punto B

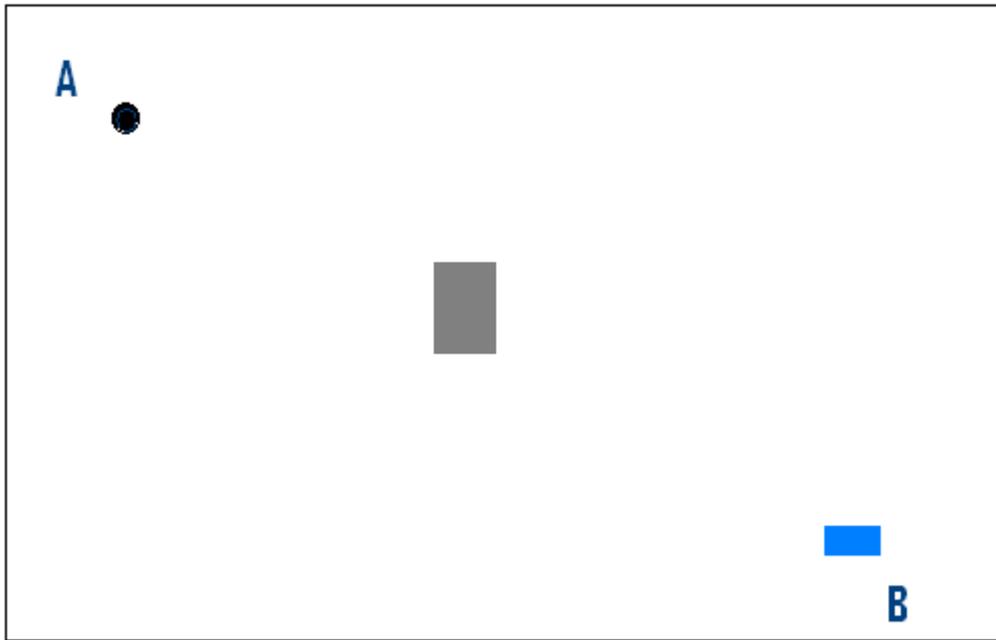


Figura V.7: Entorno de la prueba con un obstáculo, destino punto A

### 5.5.2. Resultado de la prueba

En cada uno de los ensayos el robot alcanzo su objetivo, se presentaron dos colisiones que se debieron al ángulo y la velocidad con el que el robot se acerco al obstáculo, el tiempo que le tomo para alcanzar el objetivo en cada uno de los ensayos fue variable al igual que en el caso en el que no existen obstáculos, esto debido a la variación en las medidas tomadas por el modulo GPS, que conlleva a la variación de la posición geográfica del destino en un promedio de 217 centímetros, además que el tiempo promedio para alcanzar el destino fue de 49 segundos. Los resultados de la prueba se muestran en la tabla V.5.

Ensayo	Destino alcanzado	Colisiones	Tiempo (s)	Error (cm)
1	si	si	45	80
2	si	si	60	270
3	si	no	44	110
4	si	no	48	250
5	si	no	43	250
6	si	no	55	270
7	si	no	46	120
8	si	no	50	200
9	si	no	49	150
10	si	no	54	250
PROMEDIO			49	217

Tabla V.5: Resultados prueba con un obstáculo

### 5.6. Prueba del algoritmo de navegación GPS con tres obstáculos

Esta prueba se realizó con el objetivo de conocer el comportamiento del robot móvil en un entorno con tres obstáculos entre el punto de origen y destino, con el algoritmo de de control principal que es el responsable de orientar el robot hasta el punto de destino y evitar las colisiones.

### 5.6.1. Diseño de la prueba

La prueba se llevó a cabo en un espacio abierto con tres obstáculos distribuidos en el centro del campo con una separación de 120 centímetros entre cada uno de ellos, con el robot móvil ubicado en un punto A de coordenadas  $1^{\circ} 39' 7656''$  S,  $78^{\circ} 40' 9315''$  W, con orientación contraria a la posición de destino ubicada en un punto B de coordenadas  $1^{\circ} 39' 7700''$  S,  $78^{\circ} 40' 9279''$  W, como se observa en la figura V.8 y V.9. Se considero 10 ensayos de la prueba y se determinó si se alcanzó el objetivo deseado y el tiempo que le tomo, además se determino si existieron colisiones, tomando en cuenta que se dice colisión cuando el objeto está en una posición que puede ser detectada por los sensores.

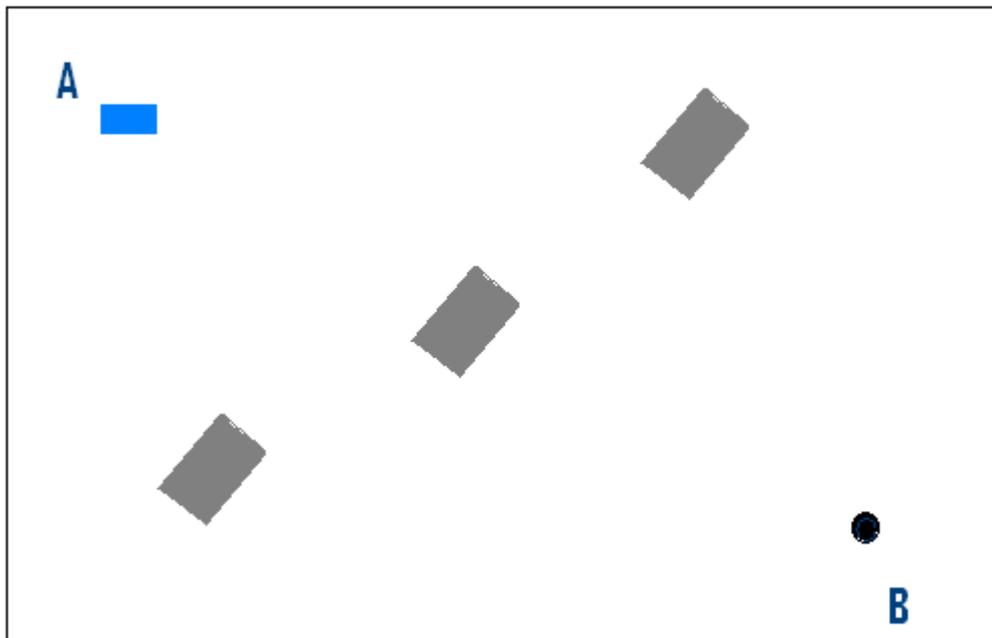


Figura V.8: Entorno de la prueba con tres obstáculos, destino punto B

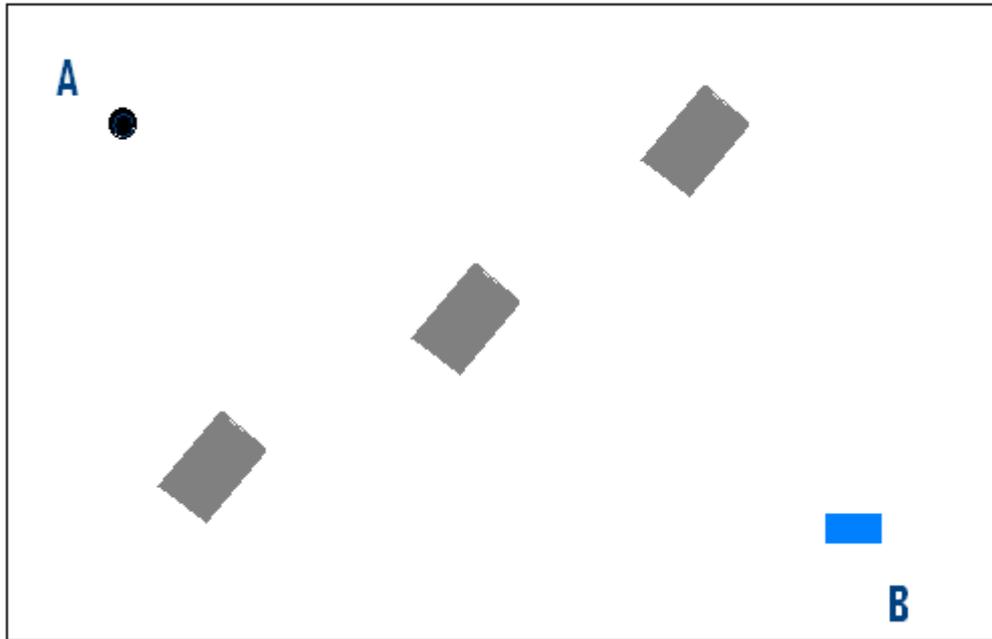


Figura V.9: Entorno de la prueba con tres obstáculos, destino punto A

### 5.6.2. Resultado de la prueba

En cada uno de los ensayos el robot alcanzó su objetivo, el tiempo máximo que le tomó al robot para alcanzar el objetivo fue de 78 segundos, pero como promedio de todos los ensayos se obtuvo un valor de 51 segundos. Al igual que en las pruebas anteriores existió una variación entre el punto de destino real y el punto que el robot alcanzó, esto se debe a la variación de las medidas del GPS que para un mismo punto entrega diferentes valores, el error máximo registrado durante las 10 pruebas fue de 400 centímetros, pero en promedio el error fue de 263 centímetros, durante todos los ensayos no existieron colisiones. Todos estos resultados se muestran en la tabla V.6.

Ensayo	Destino alcanzado	Colisiones	Tiempo (s)	Error (cm)
1	si	no	70	260
2	si	no	60	190
3	si	no	45	250
4	si	no	40	250
5	si	no	41	250
6	si	no	38	220
7	si	no	43	100
8	si	no	44	150
9	si	no	46	300
10	si	no	78	400
PROMEDIO			51	263

Tabla V.6: Resultados prueba con tres obstáculos

## 5.7. Prueba del algoritmo de navegación GPS con seis obstáculos

Esta prueba se realizó con el objetivo de conocer el comportamiento del robot móvil en un entorno con seis obstáculos entre el punto de origen y destino, con el algoritmo de de control principal que es el responsable de orientar el robot hasta el punto de destino y evitar las colisiones.

### 5.7.1. Diseño de la prueba

La prueba se llevó a cabo en un espacio abierto con seis obstáculos en el campo distribuidos con un espacio entre ellos de 120 centímetros, con el robot móvil ubicado en un punto A de coordenadas  $1^{\circ} 39' 7659''$  S,  $78^{\circ} 40' 9316''$  W, con orientación contraria a la posición de destino ubicada en un punto B de coordenadas  $1^{\circ} 39' 7704''$  S,  $78^{\circ} 40' 9280''$  W, como se observa en la figura V.10 y V.11. Se considero diez ensayos de la prueba y se determinó si se alcanzó el objetivo deseado y el tiempo que le tomo, además se determino si existieron colisiones, tomando en cuenta que se dice colisión cuando el objeto está en una posición que puede ser detectada por los sensores, para los cinco últimos ensayos se cambio el punto inicial y el final.

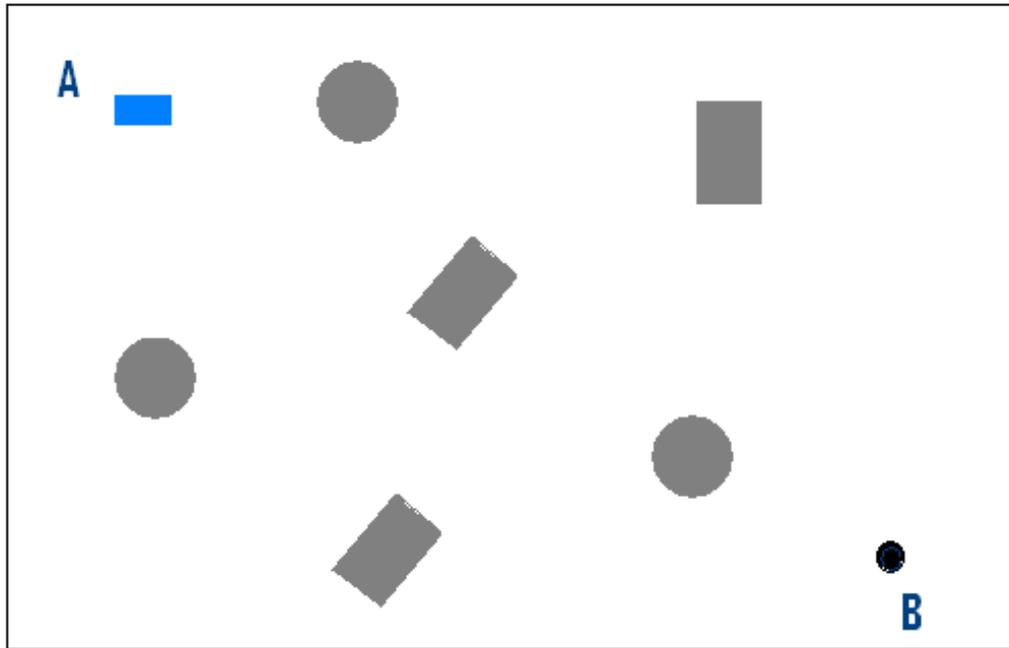


Figura V.10: Entorno de la prueba con seis obstáculos, destino punto B

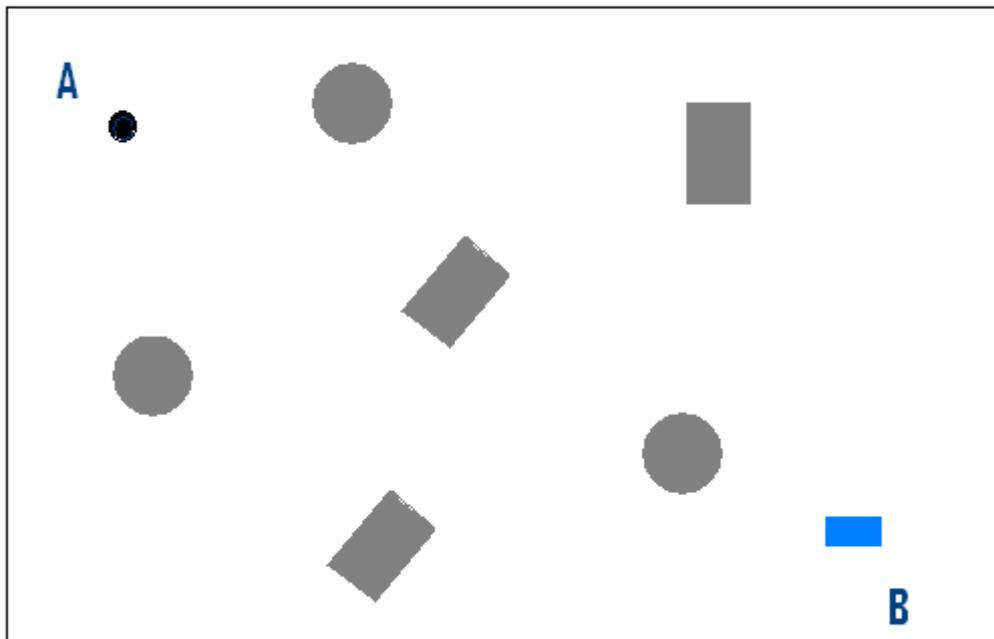


Figura V.11: Entorno de la prueba con seis obstáculos, destino punto A

### 5.7.2. Resultado de la prueba

En cada uno de los ensayos el robot alcanzo su objetivo, el tiempo máximo que le tomo al robot alcanzar el objetivo fue de 147 segundos y el error de posición máximo alcanzado fue de 500 centímetros que se tomo

al inicio de la prueba, durante el primer ensayo, esto se debió a que el numero de satélites aumento de forma repentina, con lo cual el valor entregado para el punto destino cambio notablemente, como se explico anteriormente, a mayor numero de satélites se tiene una mayor precisión de las coordenadas de un punto en la tierra.

La variación de la posición geográfica del destino tuvo un promedio de 231 centímetros y el tiempo parar alcanzar el objetivo fue de 90 segundos. Los resultados de la prueba se muestran en la tabla V.7.

Ensayo	Destino alcanzado	Colisiones	Tiempo (s)	Error (cm)
1	si	no	70	500
2	si	no	80	170
3	si	no	147	120
4	si	no	52	350
5	si	no	54	45
6	si	si	83	30
7	si	si	96	250
8	si	no	120	270
9	si	no	110	200
10	si	no	89	145
PROMEDIO			90	231

Tabla V.7: Resultados prueba con seis obstáculos

Existieron dos ensayos en los que hubo colisiones, ambas debido a que por la inercia del movimiento, el robot no alcanzó a detenerse, hay que tomar en cuenta que existe un retardo de tiempo en el que los sensores perciben los obstáculos y los microprocesadores realizan los cálculos, para tomar la decisión de esquivar los obstáculos, durante este tiempo el robot es incapaz de reaccionar rápidamente ante la presencia de un obstáculo.

## **5.8. Análisis de Resultados**

Después de culminar todas las pruebas se pudo tomar la decisión de terminar con las mejoras, el robot alcanzó a cumplir las especificaciones tomadas al inicio del proyecto y superó los entornos de pruebas diseñados, que como se vio a lo largo de este capítulo el entorno de pruebas fue creciendo en forma progresiva, es decir se comenzó en un área libre de obstáculos para finalizar en un área con 6 obstáculos. El robot es capaz de alcanzar un obstáculo en un área máxima formada por un círculo de tres metros de radio en promedio, el tiempo que le tome para alcanzar el objetivo depende claramente de la velocidad a la que se mueva y del número de obstáculos que deba tratar de sortear, tomando en cuenta que la carga de la batería influye notablemente en la velocidad a la que el robot puede moverse. Existen casos en los que el robot es incapaz de evitar los obstáculos, estos son: Cuando aparezca súbitamente al frente del robot, cuando se encuentren fuera del rango de cobertura de los sensores, cuando el ángulo de incidencia con el que se refleja la onda ultrasónica sale del cono de detección del sensor y cuando el robot se acerca a un objeto mientras está realizando los cálculos del algoritmo de evasión.

Para mayor información, en el anexo E se encuentran algunas fotografías y videos de las pruebas realizadas.

## Conclusiones

1. El sistema GPS de uso civil puede ser utilizado por cualquier persona para muchísimas aplicaciones donde se requiera conocer su ubicación en el globo terrestre.
2. El GPS al ser un sistema que trabaja en tiempo real es muy útil para implementar un robot que busca su objetivo orientándose por las coordenadas terrestres.
3. Reconocer el protocolo NMEA colaboró para decodificar los datos de los satélites y obtener las coordenadas de la ubicación terrestre del móvil.
4. La facilidad de manipular los datos por protocolos seriales de comunicación entre dispositivos hicieron que el tamaño y dificultad del hardware necesario se reduzca.
5. La brújula electrónica que es capaz de detectar el norte magnético terrestre y generar datos en grados de desviación, es el corazón de este sistema robótico móvil.
6. Con la brújula electrónica se debe tener sumo cuidado al momento de acercarse a fuentes y a materiales que absorben el campo magnético, estas producen una variación considerable en la respuesta de la brújula.
7. La elección de la fuente de alimentación adecuada influye mucho en el desempeño final del proyecto.
8. El sistema de tracción al estar desgastado por uso anterior, genera desviaciones en el trayecto pues no son simétricas en su torque, pero controladas por software son corregidas en un buen porcentaje.

9. Al probar con distintos tipos de transistores de potencia, se eligió transistores bipolares tipo Darlington por sus características en manejo de altas corrientes y bajos niveles de excitación para la implementación del controlador de los motores DC.
10. Debido a que los sensores ultrasónicos no podía convivir con sus otros similares por el efecto crosstalk, se diseñó un circuito digital que los active secuencialmente en períodos cortos para simular una actividad permanente de todos.
11. Para aplicar el método vectorial de campo de fuerzas virtuales los sensores ultrasónicos deben tener una inclinación precisa y conocida en el frente del móvil y esta es de 15 grados.
12. Los sensores ultrasónicos generan un lóbulo de radiación en su frente que en teoría recoge cualquier variación de cuerpos extraños, sin embargo por efectos de difracción cuerpos que no cumplen con cierta altura no son detectados.
13. Uno de los micro - controladores PIC usados fue elegido por sus características en memoria pues los cálculos vectoriales trigonométricos requieren de gran cantidad de variables y en consecuencia de memoria.
14. Se utilizó un sistema de entrada de datos a través de un teclado matricial y la salida se ubica en un LCD de 2 líneas y 16 caracteres, que controlados por software permiten que el robot funcione en modo solo lectura y ejecución.
15. El sistema robótico admite un funcionamiento autónomo de 30 minutos debido a las necesidades de corriente de la etapa de potencia.

16. El sistema GPS introduce un error en la lectura de la posición global de aproximadamente 2 metros de radio por lo que al alcanzar el objetivo se toma en cuenta este rango de error.
17. Las ruedas de goma en la estructura de triciclo funcionan bien en la mayoría de entornos y reduce las vibraciones.
18. Al no tener sistemas de amortiguamiento el móvil sufre daños en terrenos irregulares.
19. El poderoso algoritmo de cálculo permite resolver el rumbo que ha de tomar el móvil en tiempo real leyendo constantemente los datos del módulo GPS.
20. El sistema de detección de obstáculos tiene un tiempo de respuesta lento debido a la limitada velocidad de procesamiento de los micro controladores.
21. El sistema operativo del robot permite la gestión de todos los módulos por separado con fines de monitoreo.
22. La navegación GPS del robot móvil detector de obstáculos funciona correctamente bajo condiciones controladas en un entorno prediseñado.
23. Se adaptó el algoritmo de fuerzas virtuales a una forma más sencilla que se pueda programar para un micro-controlador PIC, la eficacia del algoritmo depende mucho de la precisión y la cantidad de los sensores ultrasónicos utilizados.
24. Existe un tiempo entre la lectura de los sensores y el cálculo del algoritmo de navegación y evasión de obstáculos, en el que el móvil

es incapaz de percibir su entorno por lo que se pueden presentar colisiones.

### **Campos de aplicación**

El sistema de navegación por GPS con detección de obstáculos puede ser útil en áreas peligrosas donde se necesite realizar inspecciones o recolección de materiales pero la integridad humana se vea amenazada, así también en sistemas de vigilancia donde se puede limitar el recorrido del móvil con coordenadas terrestres y mantener vigilada una determinada área o recinto. Se puede implementar el sistema en móviles aéreos para realizar avistamientos desde la altura, de torres de antenas, vigilancia de fronteras o recolección de información del clima.

### **Trabajos a futuro**

El móvil robótico puede mejorar sus prestaciones agregándole más sensores laterales para que sea capaz de registrar y moverse paralelo a paredes, también se puede mejorar la navegación con una estructura mecánica mas adaptable a terrenos extremos como patas u orugas. Si tomamos en cuenta la duración de la autonomía también se sugiere implementar un mejor sistema de alimentación basado en baterías de polímero de litio que reducirán el peso total del móvil e incrementará su tiempo de actividad independiente, considerando para esto la seguridad implícita en el manejo de este tipo de baterías.

## RECOMENDACIONES

1. La batería del robot es de plomo y funciona con ácido sulfúrico, ambos componentes son muy tóxicos, evitar el contacto con la piel.
2. Cuando se recarga la batería, el ácido dentro reacciona burbujeando y produce gas que no debe ser inhalado, así mismo evitar que el ácido salpique las partes del robot para no producir corrosiones.
3. Luego de cada sesión de navegación se debe conectar la batería a su cargador para mantener en buen estado la química interior de la misma.
4. El robot no tiene sistema de amortiguación, no se deben realizar sesiones de navegación en terrenos accidentados para no romper las placas ni componentes.
5. La brújula para un óptimo rendimiento debe estar posicionada horizontalmente respecto al plano del robot, de lo contrario se producirán errores de lectura en el ángulo de rumbo.
6. La etapa de potencia tiene un disipador de calor incorporado, sin embargo evitar el sobrecalentamiento con excesivo tiempo de funcionamiento para evitar la destrucción de los transistores.
7. El escenario de navegación no debe ser modificado cuando el robot se encuentra en funcionamiento.
8. Para un mejor desempeño en terrenos más generales, recomendamos modificar el sistema de tracción a uno de cuatro ruedas o de sistema oruga.

9. Con el fin de mejorar los tiempos de respuesta se recomienda buscar micro-controladores que trabajen a mayor velocidad.
  
10. Antes de iniciar una prueba el robot necesita leer un mínimo de 8 satélites para minimizar el rango de error en el punto de llegada.
  
11. El ingreso de coordenadas no tiene control de errores, debido a la falta de memoria por tanto evitar el ingreso erróneo de coordenadas

## RESUMEN

El presente trabajo de tesis desarrolló el sistema de navegación por GPS de un robot móvil detector de obstáculos, con el objetivo de determinar la fiabilidad de la navegación autónoma por GPS.

El móvil se montó en aluminio, el hardware está constituido por dos motores DC, procesador de datos con microcontroladores PIC, 4 sensores ultrasónicos, compás electrónico, módulo GPS y batería de 12V a 7 amperios. Utilizando método deductivo - experimental se acoplaron las etapas desarrollando interfaces de comunicación siguiendo estándar NMEA 0183 y protocolos RS232 e I2C. El software se escribió en Microcode y MicroBasic, usando el método de llamado de procedimientos y funciones.

Con el software depurado y programado en microcontroladores, se inició la fase de pruebas en un escenario de dimensiones 20 x 30 metros, con obstáculos cúbicos de 50 cm, realizándose 10 intentos obteniendo 10% de éxito al alcanzar el objetivo y 25% esquivando obstáculos, tras la re-calibración del compás, re-ubicación del módulo sonar y corrección de procedimientos de software involucrados, se repitieron otras 10 pruebas y se obtuvo el 100% alcanzando el punto de llegada con 5 metros de error de precisión, 80% al esquivar obstáculos, diferencia de tiempo de ejecución entre 1 y 2 minutos variando los puntos de partida y llegada y 30 minutos de autonomía en el total de las pruebas.

Se comprobó la fiabilidad de navegación por GPS del prototipo, que funcionó según los parámetros establecidos al inicio del proyecto.

## SUMMARY

The present thesis work developed the navigation system by GPS of an obstacle detector mobile robot to determine the automate navigation reliability by GPS.

The mobile device was mounted in aluminum, the hardware consists of two DC motors, data processors with micro-controllers PIC, 4 ultrasonic sensors, electronic compass, GPS module and 12V battery at 7 amperes. Using the deductive-experimental method the steps were matched developing communication interphases according to the standard NMEA 0183 and Protocols RS232 and I2C. The software was written in Microcode and MicroBasic, using the procedure and function call method.

With the purified and programmed in micro-controllers software, the test phase was initiated in a stage of 20X30 meters, with 50cm cubic obstacles, carrying out 10 trials with 10% success upon attaining the objective and 25% avoiding obstacles. After having rated the compass, replacing the sonar module and correcting the involved procedures, other 10 trials were repeated and 100% was obtained reaching the arrival point with 5 meters precision error, 80% upon avoiding obstacles, an execution time difference from 1 to 2 minutes varying the starting and arrival points and 30 minutes autonomy in all tests.

The navigation reliability by the GPS of the prototype which functioned according to the established parameter at the project beginning was tested.

## GLOSARIO

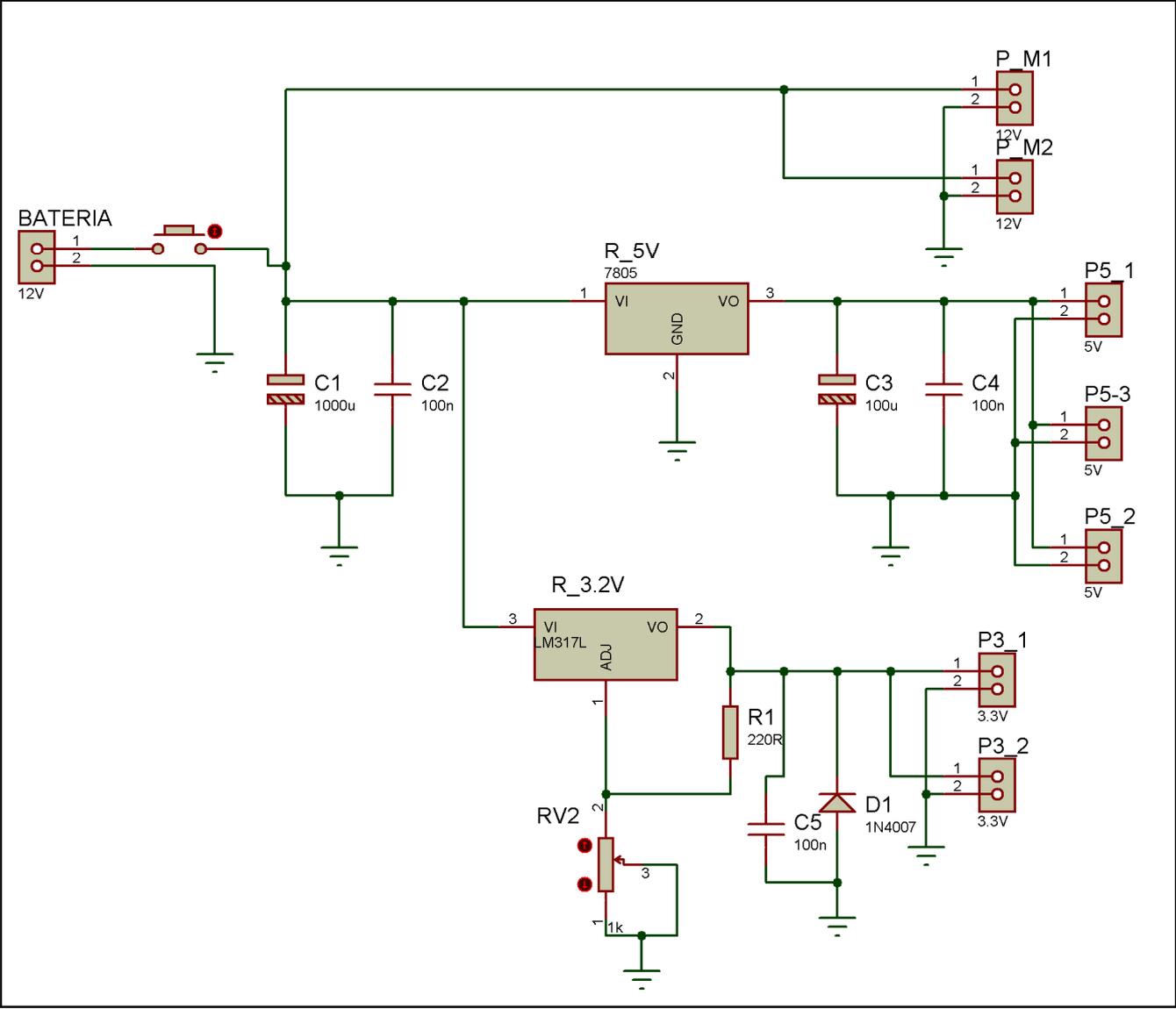
### **ASCII**

<b>CA/Code.</b>	Coarse/Acquisition- Code.
<b>CR.</b>	Retorno de Carro
<b>DC.</b>	Corriente Directa
<b>DOP.</b>	Factor de Dilución de Precisión
<b>EGM.</b>	Modelo de Campo Gravitacional de la Tierra
<b>GGA.</b>	Global Positioning System Fix Data
<b>GLL.</b>	Geographic Position – Latitude/Longitude
<b>GNSS.</b>	
<b>GPS.</b>	Sistema de Posicionamiento Global.
<b>GSV.</b>	Satellites in view (Satelites a la Vista)
<b>I2C.</b>	
<b>LF.</b>	Línea de Alimentación
<b>MCS.</b>	Master Control Station (Estación de Control Maestra).
<b>NMEA.</b>	Asociación Nacional de Electrónica Marina
<b>OEM.</b>	
<b>OCS.</b>	Operational Control System (Sistema Operacional de Control).
<b>P-Code.</b>	Precision-Code.
<b>PPS.</b>	Precise Positioning Service (Servicio de Posicionamiento Preciso).
<b>PRN.</b>	Pseudo-Random Noise (Ruido Pseudo-Aleatorio)
<b>PW.</b>	Anchura de Pulso
<b>PWM.</b>	Pulse Width Modulation (Modulación por Ancho de Pulso).
<b>RMB.</b>	Recommended Minimum Navigation Information (Información de Navegación Mínima Recomendada)
<b>SCL.</b>	Señal de Reloj I2C
<b>SDA.</b>	Señal de Datos I2C
<b>SMP.</b>	
<b>SPS.</b>	Standard Positioning Service (Servicio de Posicionamiento Estándar).
<b>SSM.</b>	Spread Spectrum Modulation (Modulación de Espectro Ensanchado).

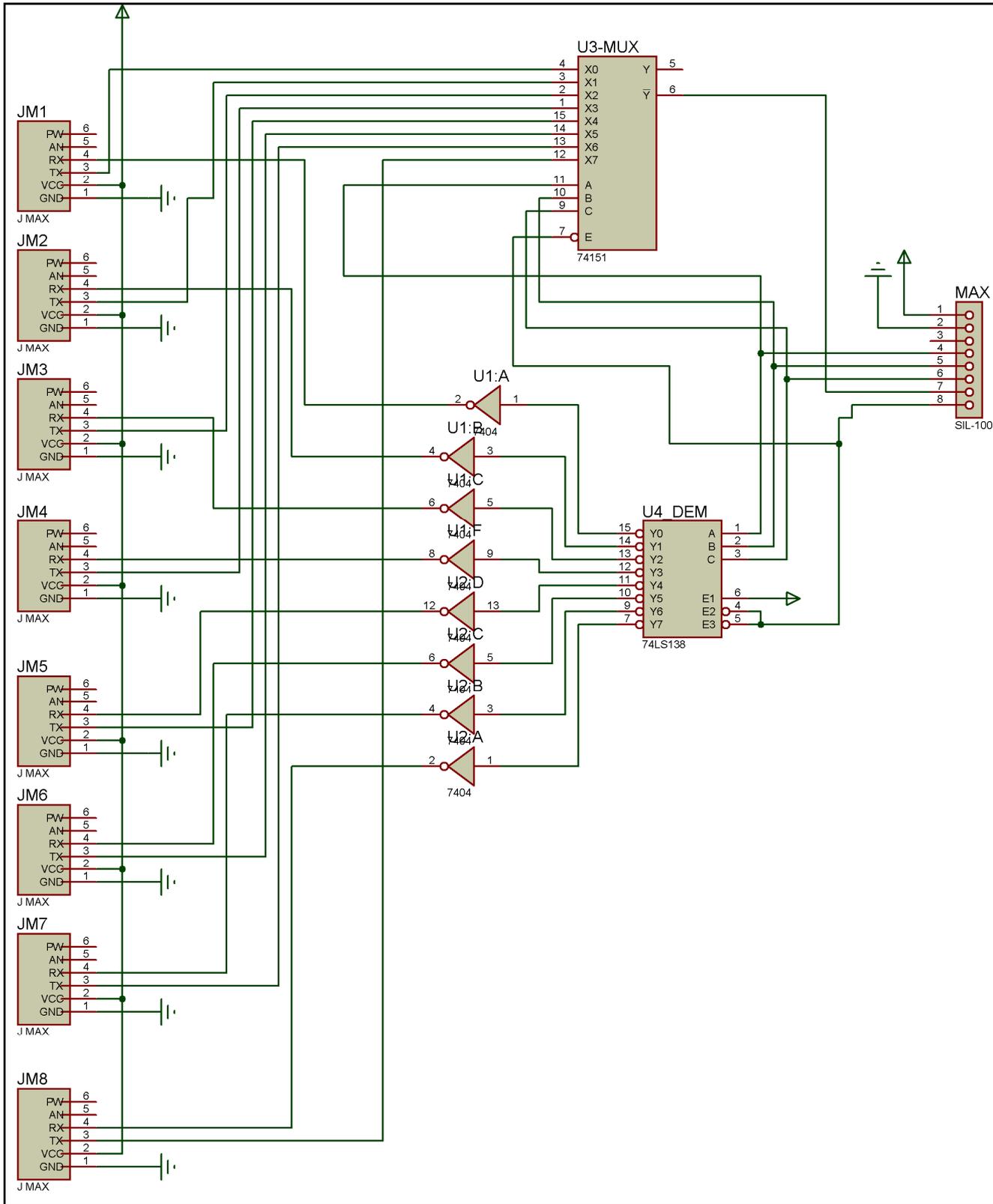
**SSP.** Puerto Serie Síncrono  
**TOF.** Tiempo de Vuelo  
**USART / SCI.**  
**UTM.** Marcador transversal universal.  
**VFF.** Virtual Force Field (Campo de Fuerzas Virtuales)  
**VTG.** Track Made Good and Ground Speed  
**WSG84.** Sistema coordinado geodésico mundial de 1984.

## **ANEXOS**

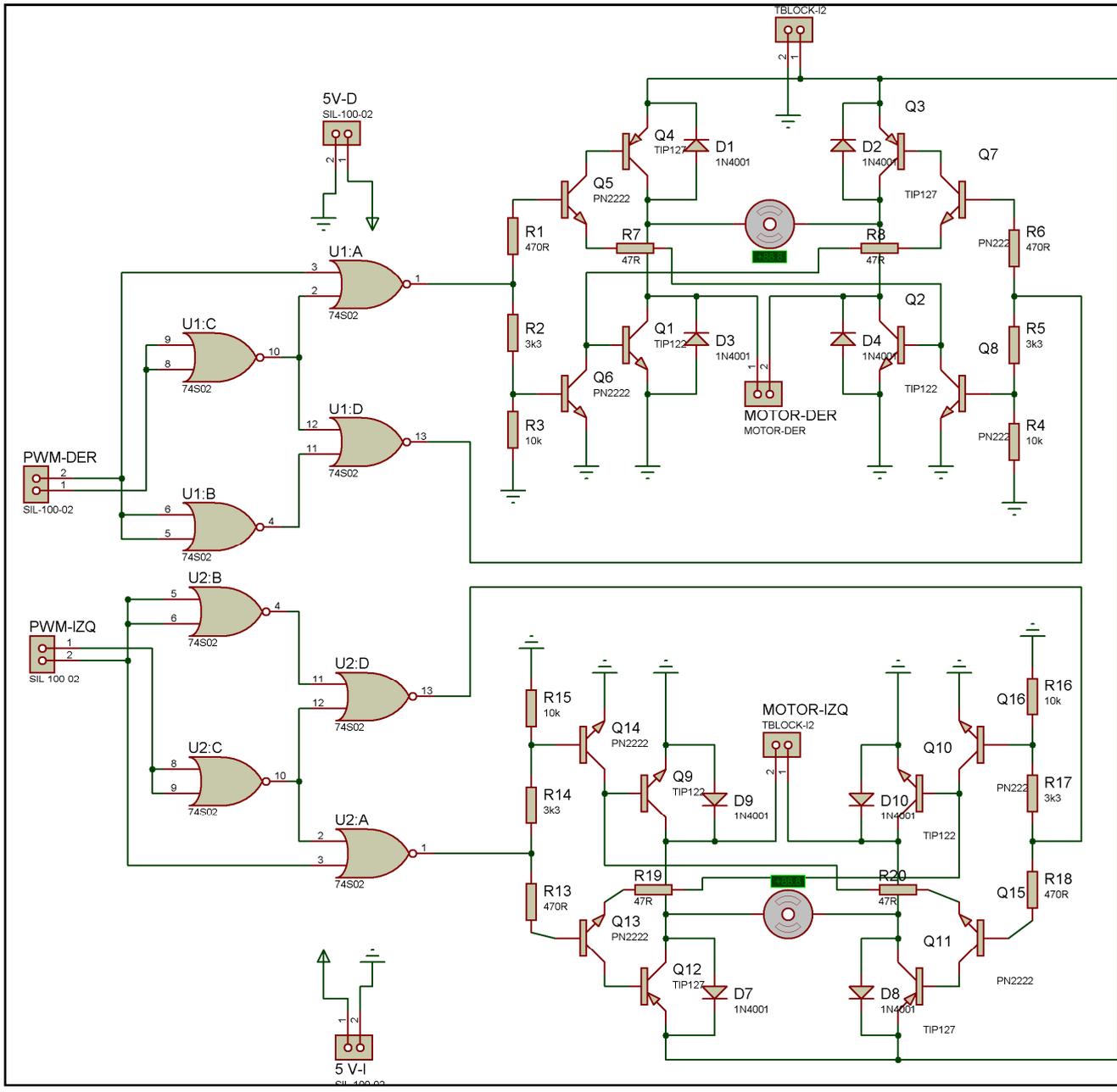
ANEXO B: DIAGRAMAS Y CIRCUITOS IMPRESOS



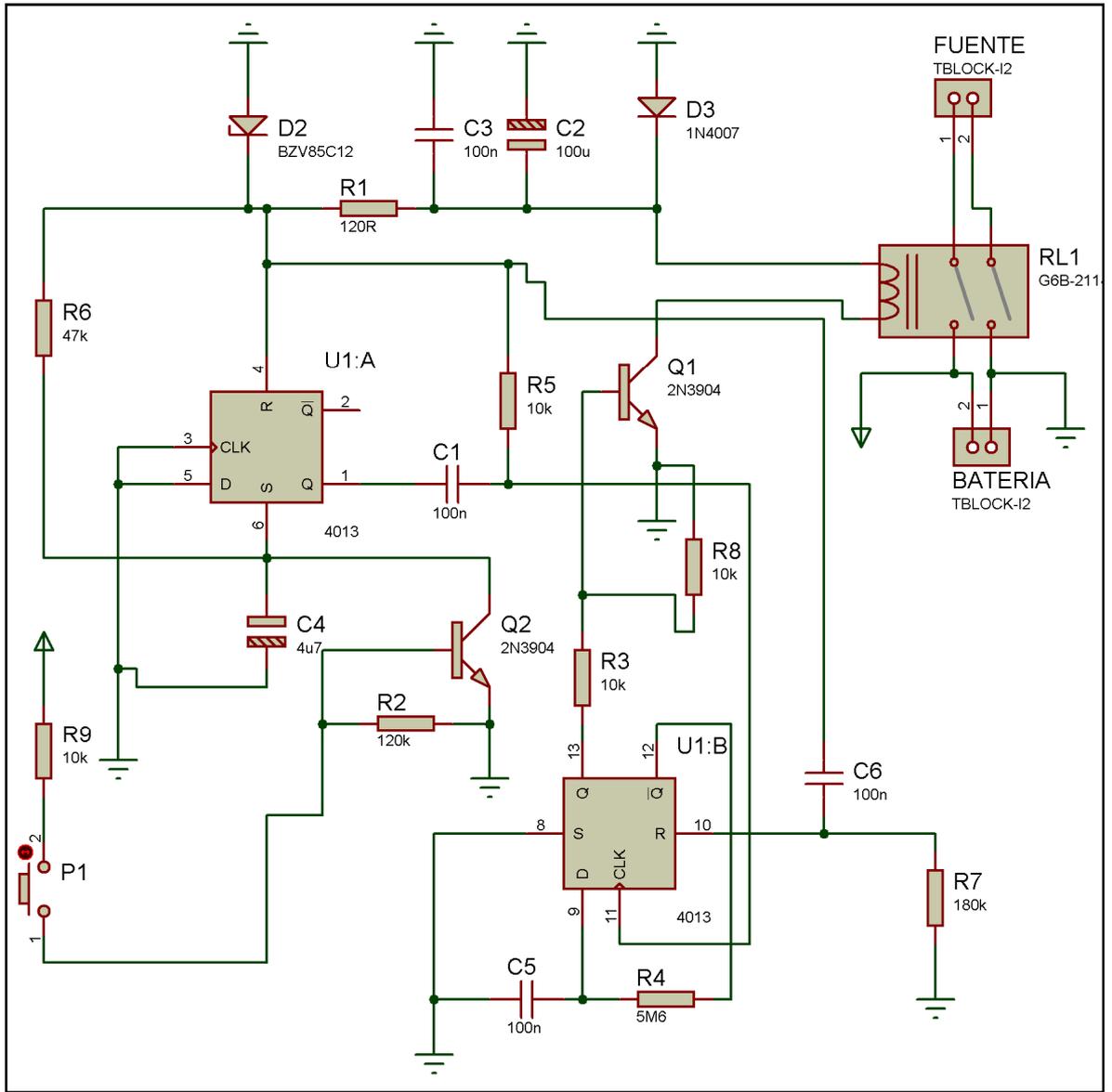
B.1. Esquema electrónico Fuente de Alimentación



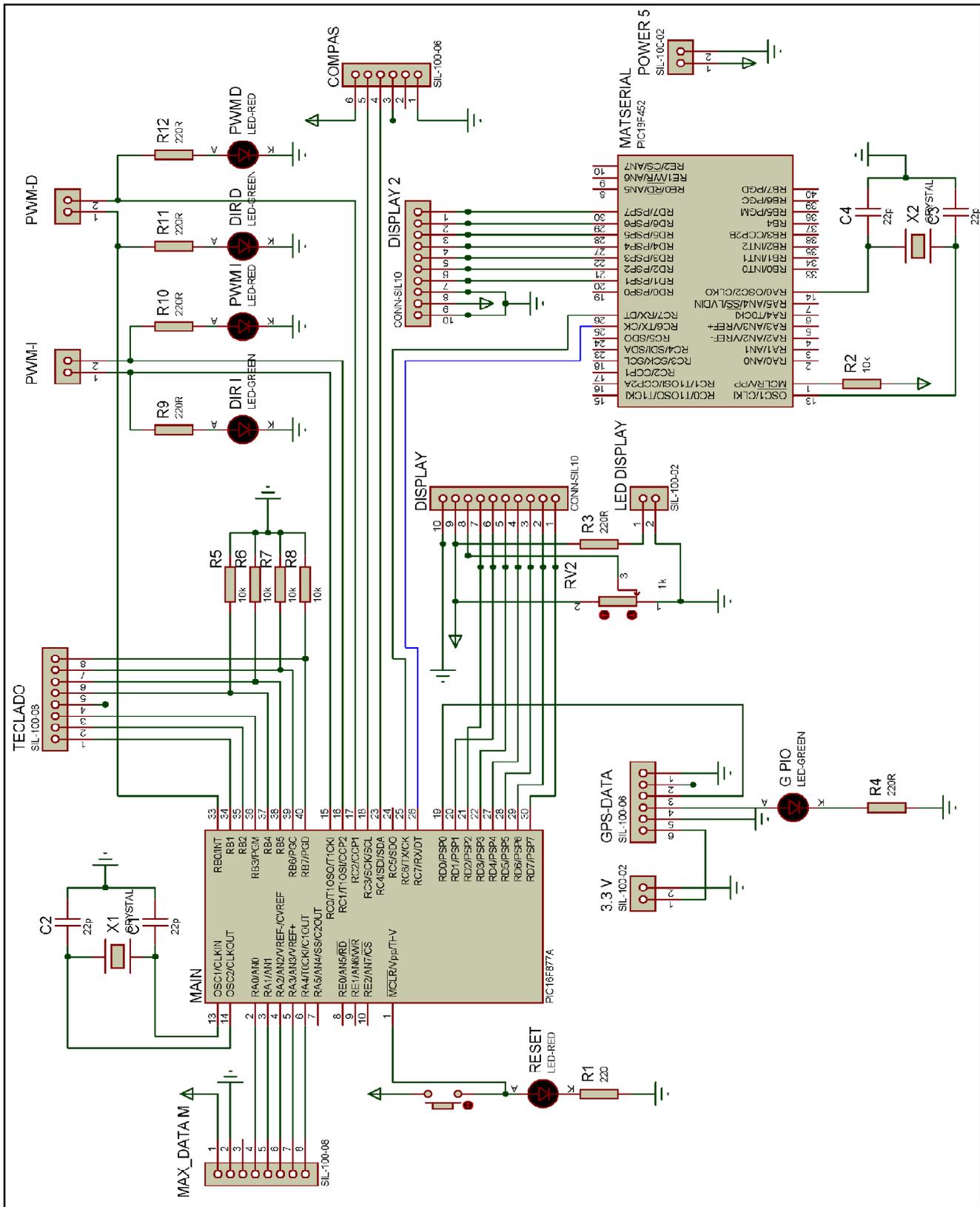
B.2. Esquema electrónico Modulo Sonar



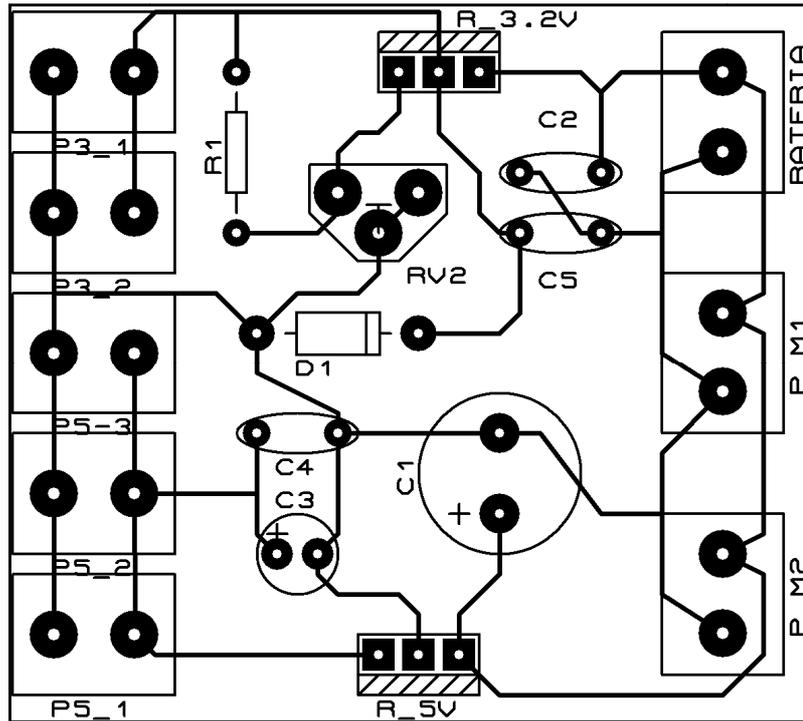
B.3. Esquema electrónico Driver Puente H



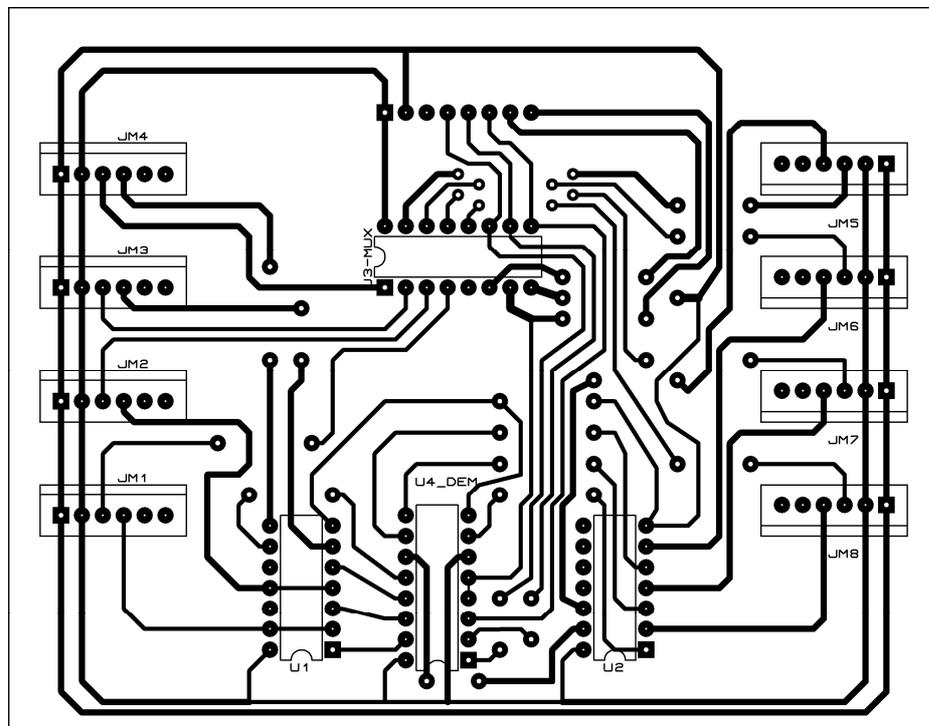
B.4. Esquema electrónico Interrupción digital



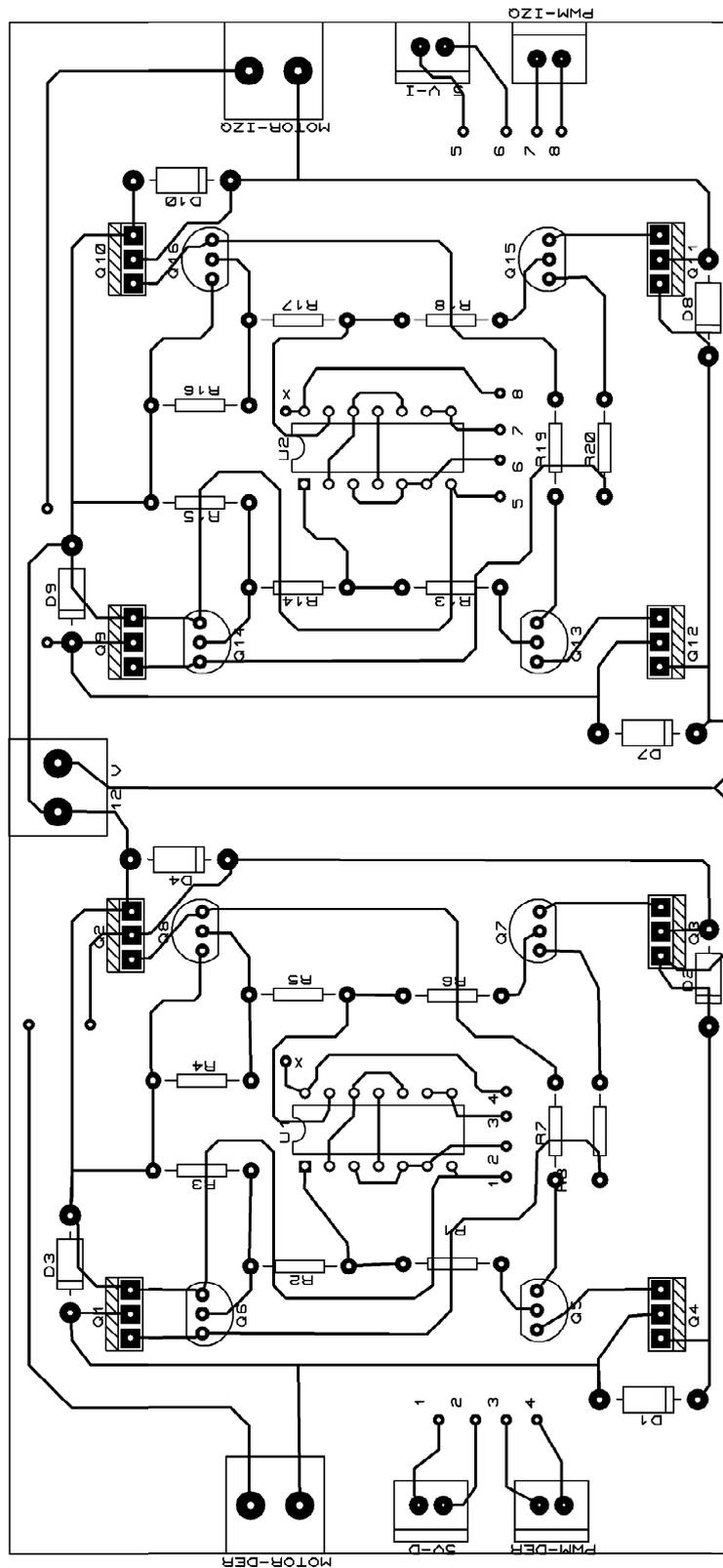
B.5. Esquema electrónico Placa principal



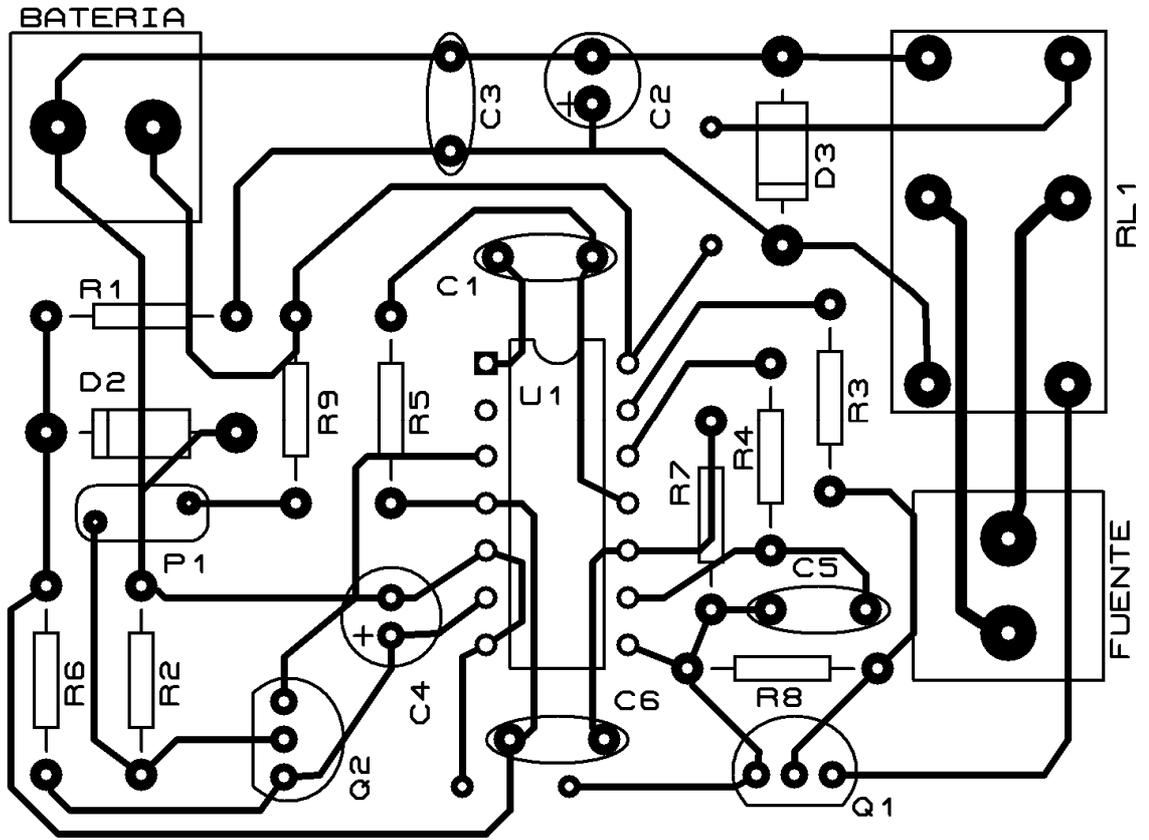
B.6. Circuito Impreso Fuente de Alimentación



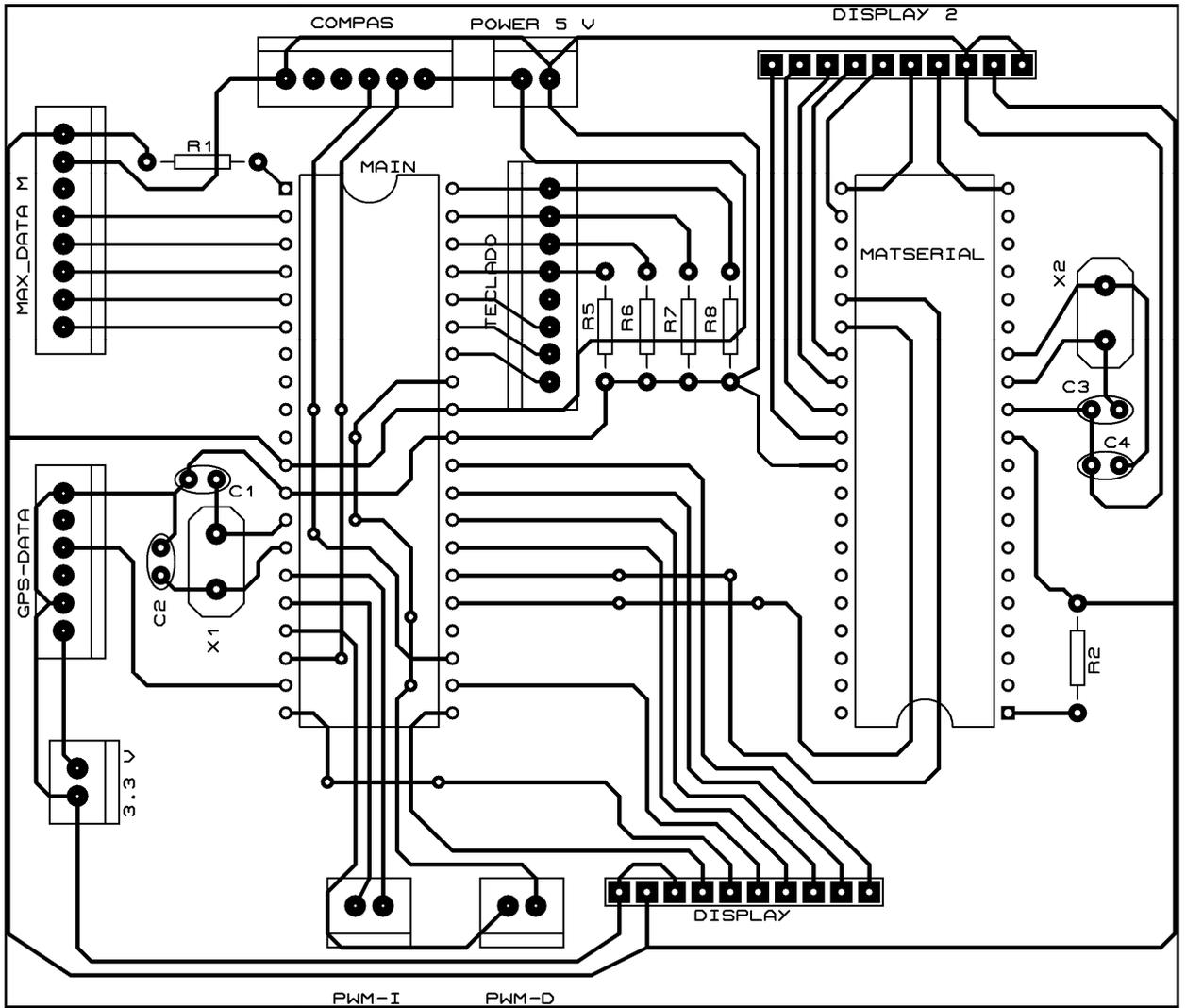
B.7. Circuito Impreso Modulo Sonar



B.8. Circuito Impreso Driver Puente H



B.9. Circuito Impreso Interruptor electrónico



B.10. Circuito Impreso Placa Principal

## ANEXO C: CODIGO FUENTE PROGRAMA PRINCIPAL

\*\*\*\*\*

'\* Name : PROGRAMA PRINCIPAL.BAS \*

'\* Author : Geovanny Simba, Paúl Aguilar \*

'\* Notice : Copyright (c) 2008 GEOSIM \*

'\* : All Rights Reserved \*

'\* Date : 10/10/2008 \*

'\* Version : 1.0 \*

'\* Notes : \*

\*\*\*\*\*

*'INICIO DEL DISPLAY*

-----

*'Selección Puerto de Datos LCD*

DEFINE LCD\_DREG PORTD

*'Selección cabecera 4 bits (0 para bus de 8 bits)*

DEFINE LCD\_DBIT 4

*'Selección de puerto de registro LCD*

DEFINE LCD\_RSREG PORTD

*'Selección del bit de registro select LCD*

DEFINE LCD\_RSBIT 2

*'Selección puerto Enable LCD*

DEFINE LCD\_EREG PORTD

*'Selección del bit Enable LCD*

DEFINE LCD\_EBIT 3

*'selección tamaño del bus LCD*

DEFINE LCD\_BITS 4

*'Selección del numero de líneas en e LCD*

DEFINE LCD\_LINES 2

*'Selección del comando delay en us*

DEFINE LCD\_COMMANDUS 2000

*'Selección de retardo de datos en us*

DEFINE LCD\_DATAUS 40

DEFINE LCD\_RWREG PORTD *' Puerto del pin read/write del LCD*

DEFINE LCD\_RWBIT 1 *' Bit de pin read/write del LCD*

-----  
*'PARAMETROS PUERTO SERIAL COMUNICACIÓN SENSORES ULTRASONICOS  
MAXSONAR*

-----  
BAUDMAX VAR BYTE

MAXSin VAR PORTA.4 *'entrada de datos*

baudMAX = 16468 *'9600,8N1 con MAX232*

-----  
*'VARIABLES MAXSONAR.....*

-----  
ADCON1=%110111

TRISA = %110000 *'RA0-RA3 salida, RB4-RB5 entrada*

TRISE = %100 *'RE0-RE1 salida, RE2 entrada*

J VAR BYTE

CONT\_ZERO VAR BYTE *'contador de ceros*

CONT\_ONE VAR BYTE *'contador de unos*

PROMEDIO VAR WORD

MAX\_LI VAR WORD *'sensor lateral izquierdo*

MAX\_LD VAR WORD *'sensor lateral derecho*

MAX\_CI VAR WORD *'sensor central izquierdo*

MAX\_CD VAR WORD *'sensor central derecho*

MAX\_RES VAR byte[4]

MAX\_I VAR BYTE

-----  
*'PARAMETROS TECLADO*

```

-----
OLDRESULT VAR BYTE           'carácter previo
KEYPRESS VAR BYTE           'tecla presionada
RESULT VAR BYTE             'carácter leído
FLAG VAR BIT                'temporal
RESUL VAR BYTE
SHIFT VAR BYTE              'cambio función
TRISB=%11110000            'RB0-RB3 salida, RB4-RB7 entrada

```

```

-----
'PARAMETROS PUERTO SERIAL GPS

```

```

-----
BAUDGPS VAR BYTE
GPSin VAR PORTD.0           ; 'entrada de datos
baudGPS = 188 ;4800,8N1 CON MAX232

```

```

-----
'DEFINICION DE VARIABLES GPS

```

```

-----
GPS_DIA var Byte           ' día
GPS_MES var Byte          ' mes
GPS_ANIO var Byte         ' año
GPS_HORAS var Byte       ' horas
GPS_MINUTOS var Byte     ' minutos
GPS_SEGUNDOS var Byte    ' segundos
GPS_LAT_GRADOS var Byte  '\
GPS_LAT_MINUTOS var Byte '\
GPS_LAT_SEGUNDOS var byte '/ coordenadas de latitud
GPS_LAT_mSEGUNDOS var byte '/'
GPS_LAT_DIRECCION var Byte '/'
GPS_LON_GRADOS var Byte  '\
GPS_LON_MINUTOS var Byte '\

```

```

GPS_LON_SEGUNDOS var byte           ' / coordenadas de longitud
GPS_LON_mSEGUNDOS var byte         ' /
GPS_LON_DIRECCION var Byte         ' /
GPS_CURSO_GROUND VAR BYTE         ' curso sobre la tierra
GPS_SPEED_GROUND VAR BYTE         ' velocidad sobre la tierra
GPS_MAGNETIC_VARIATION VAR BYTE   ' variación magnética
GPS_FIX_TYPE var Byte              ' tipo de arreglo
GPS_SATS_IN_VIEW var Byte         ' numero de satélites a la
vista

```

```

GP_LATS var Byte[13]              'vectores de
almacenamiento

```

```

GP_LONS var Byte[12]

```

```

GP_LATD var Byte[13]

```

```

GP_LOND var Byte[12]

```

```

GPS_COURSE var WORD

```

```

GPS_MAGVAR var BYTE

```

```

GPS_RX var Byte[85]              ' buffer serial para sentencias
NMEA

```

```

I var byte

```

```

-----

```

```

'PARAMETROS PUERTO SERIAL TX_RX 18F

```

```

-----

```

```

BAUDP18 VAR BYTE

```

```

P18in VAR PORTC.7                ; entrada de datos

```

```

P18out VAR PORTC.6               ; Salida hacia la PIC
16F877A

```

```

baudP18 = 84                      ;9600,8N1 con MAX232

```

```

-----

```

```

'VARIABLES RUMBO PIC 18.....

```

-----  
RUMBO VAR WORD

VELOCIDAD VAR byte

VELOCIDAD1 VAR byte  
-----

*'PARAMETROS CONTROL PWM Y DIRECCION*  
-----

DEFINE CCP1\_REG PORTC *'Hpwm 1 puerto del pin*

DEFINE CCP1\_BIT 2 *'Hpwm 1 bit del pin*

DEFINE CCP2\_REG PORTC *'Hpwm 2 puerto del pin*

DEFINE CCP2\_BIT 1 *'Hpwm 2 bit del pin*

SYMBOL DIR\_I=PORTC.0 *'Señal Adelante/Atrás motor*  
*izquierdo*

SYMBOL DIR\_D=PORTB.0 *'Señal Adelante/Atrás motor*  
*derecho*

PWM\_I CON 2 *'Señal PWM motor izquierdo*

PWM\_D CON 1 *'Señal PWM motor derecho*

DIR VAR BYTE  
-----

*'PARAMETROS BRUJULA CMP03*  
-----

DPIN var PORTc.4 *' Pin de datos I2C*

CPIN var PORTc.3 *' Pin de reloj I2C*

bearing var word *' Valor del ángulo entre 0-360*  
*grados*  
-----

*'VARIABLES DE COMPARACIÓN DESTINO ALCANZADO*  
-----

LGD VAR BYTE *'Latitud de destino en grados*

LMD VAR BYTE *'Latitud de destino en minutos*

LSD VAR BYTE *'Latitud de destino en segundos*

LSSD VAR BYTE *'Latitud de destino en milésimas de segundo*

GGD VAR BYTE	<i>'Longitud de destino en grados</i>
GMD VAR BYTE	<i>'Longitud de destino en minutos</i>
GSD VAR BYTE	<i>'Longitud de destino en segundos</i>
GSSD VAR BYTE	<i>'Longitud de destino en milésimas de segundo</i>
LGS VAR BYTE	<i>'Latitud de salida en grados</i>
LMS VAR BYTE	<i>'Latitud de salida en minutos</i>
LSS VAR BYTE	<i>'Latitud de salida en segundos</i>
LSSS VAR BYTE	<i>'Latitud de salida en milésimas de segundo</i>
GGs VAR BYTE	<i>'Longitud de salida en grados</i>
GMS VAR BYTE	<i>'Longitud de salida en minutos</i>
GSS VAR BYTE	<i>'Longitud de salida en segundos</i>
GSSS VAR BYTE	<i>'Longitud de salida en milésimas de segundo</i>

-----

*'INICIO DE VARIABLES*

-----

**INICIO:**

GOSUB PARAR	<i>'PWM en STOP, motores parados</i>
PAUSE 2000	<i>'tiempo de inicio GPS</i>
PORTA=%00001000	<i>'todos los max-zonar parados</i>

-----

*'INICIO PROGRAMA*

-----

```
Lcdout $FE, 1
LCDOUT $FE, $0F
LCDOUT "NAV-SAT V 1.0"
Lcdout $FE, $C0
LCDOUT "INICIANDO.."
PAUSE 50
LCDOUT ".."
PAUSE 50
```

LCDOUT ".."

PAUSE 50

LCDOUT ".."

PAUSE 50

LCDOUT ".."

PAUSE 50

-----  
'MENÚ INDIRECTO  
-----

REPEAT

PORTB=%1000

IF PORTB.4=1 THEN

*'Lectura modulo GPS*

LCDOUT \$FE, 1

LCDOUT \$FE, \$0F

LCDOUT "ESPERANDO SATEL"

REPEAT

GOSUB EXTRAC\_COOR\_GPS

GOSUB LCD\_GPS\_COOR

UNTIL(PORTB.7=1)

PAUSE 800

GOTO INICIO

ENDIF

IF PORTB.5=1 THEN

*'Lectura brújula CMP03*

REPEAT

GOSUB CMP03

UNTIL(PORTB.7=1)

PAUSE 800

GOTO INICIO

ENDIF

IF PORTB.6=1 THEN

*'Lectura Modulo Sonar*

```

        REPEAT
            GOSUB MAX_MAIN
            GOSUB MAX_LCD
        UNTIL(PORTB.7=1)
        PAUSE 800
        GOTO INICIO
    ENDIF
UNTIL(PORTB.7=1)
'-----
'INGRESO DEL DESTINO
'-----

    GOSUB MAIN_KEYPAD
    PORTB=%1000
    GOSUB MAIN
'-----
'PROGRAMA PRINCIPAL
'-----

MAIN:
    if PORTB.4=1 thenN                                'Reinicio del sistema
        PAUSE 500
        GOTO INICIO
    endif
    GOSUB MAX_MAIN
    gosub CMP03
    GOSUB EXTRAC_COOR_GPS
    gosub DSEGURIDAD
    GOSUB PARAR
    GOSUB TX_RX_18F
    gosub DIRECCION
'-----

```

*'COMPARACIÓN PUTO ACTUAL Y DESTINO*

-----  
IF (LSD=LSS)and (GSD=GSS) THEN  
  IF (GSSS>(GSSD-4)) AND (GSSS<(GSSD+4)) THEN  
    IF (LSSS>(LSSD-4)) AND (LSSS<(LSSD+4)) THEN  
      GOSUB PARAR  
      LCDOUT \$FE, 1  
      LCDOUT \$FE, 2  
      LCDOUT "OBJET. ALCANZADO!!"  
      PAUSE 2000  
      GOTO INICIO  
    ENDIF  
  ENDIF  
ENDIF  
GOTO MAIN  
END

-----  
*'EXTRACCION DE ANGULO DE LA BRUJULA CMP03*  
-----

**CMP03:**

I2CREAD  
DPIN,CPIN,%11000000,2,[BEARING.HIGHBYTE,BEARING.LOWBYTE]  
  BEARING.HIGHBYTE=BEARING.HIGHBYTE-128  
  LCDOUT \$FE, 2  
  LCDOUT "Bearing A: ",DEC BEARING,"     "  
RETURN

*'ENVIO DE COORDENADAS AL PIC 18F\*\*\*\**  
-----

**TX\_RX\_18F:**

PAUSE 1  
serout2 P18out,baudP18,["PS",GP\_LATS[2],GP\_LATS[3],GP\_LATS[4]

```
,GP_LATS[5],GP_LATS[6],GP_LATS[7],GP_LATS[8],GP_LATS[9]_  
,GP_LATS[10],"OK"]           'Latitud de salida
```

PAUSE 1

```
serout2 P18out,baudP18,[GP_LONS[0],GP_LONS[1],GP_LONS[2]_  
,GP_LONS[3],GP_LONS[4],GP_LONS[5],GP_LONS[6],GP_LONS[7]_  
,GP_LONS[8],GP_LONS[9],"OK"]           'Longitud de salida
```

PAUSE 1

```
serout2 P18out,baudP18,["PD",dec GP_LATD[2],dec GP_LATD[3],dec  
GP_LATD[4],dec GP_LATD[5],dec GP_LATD[6],dec GP_LATD[7],dec_  
GP_LATD[8],dec GP_LATD[9],GP_LATD[10],"OK"]   'Latitud de destino
```

PAUSE 1

```
serout2 P18out,baudP18,[dec GP_LOND[0],dec GP_LOND[1],dec  
GP_LOND[2]_  
,dec GP_LOND[3],dec GP_LOND[4],dec GP_LOND[5],dec GP_LOND[6]_  
,dec GP_LOND[7],dec GP_LOND[8],GP_LOND[9],"OK"] 'Longitud de  
destino
```

PAUSE 1

```
serout2 P18out,baudP18,[dec MAX_LI,"OK"]
```

PAUSE 1

```
serout2 P18out,baudP18,[dec MAX_cl,"OK"]
```

pAUSE 1

```
serout2 P18out,baudP18,[dec MAX_Cd,"OK"]
```

PAUSE 1

```
serout2 P18out,baudP18,[dec MAX_Ld,"OK"]
```

PAUSE 1

```
serout2 P18out,baudP18,[dec BEARING,"OK"]
```

SerIn2

```
P18in,baudP18,2000,PIC_TIMEOUT,[wait("RM"),GPS_RX[0],GPS_RX[1]_
```

```
,GPS_RX[2],GPS_RX[3],GPS_RX[4],GPS_RX[5],GPS_RX[6],GPS_RX[7]_
```

```
,GPS_RX[8]]           'Recepción de datos algoritmo de evasión
```

```
RUMBO=(GPS_RX[0]-48)*1000+(GPS_RX[1]-48)*100+_
      (GPS_RX[2]-48)*10+(GPS_RX[3]-48)
DIR=(GPS_RX[4]-48)*10+(GPS_RX[5]-48)
VELOCIDAD1=(GPS_RX[6]-48)*100+(GPS_RX[7]-48)*10+(GPS_RX[8]-48)
LCDOUT $Fe, $C0,"Rm:" ,DEC RUMBO,"",DEC (GPS_RX[4]-48)_
      ",", DEC (GPS_RX[5]-48),",", DEC VELOCIDAD1
```

```
RETURN
```

```
'-----
'LIMITE DE SEGURIDAD (DISTANCIA A UN OBJETO)
'-----
```

**DSEGURIDAD:**

```
if ((MAX_CD<40) OR (MAX_CI<40)) THEN
```

```
  GOSUB PARAR
```

```
  IF MAX_LI>80 THEN
```

```
    GOSUB IZQUIERDA
```

```
    GOTO MAIN
```

```
  ELSE
```

```
    IF MAX_LD>80 THEN
```

```
      GOSUB DERECHA
```

```
      GOTO MAIN
```

```
    ELSE
```

```
      GOSUB ATRAS
```

```
      GOTO MAIN
```

```
    ENDIF
```

```
  ENDIF
```

```
ENDIF
```

```
RETURN
```

```
'-----
'CONTROL DE DIRECCION Y POTENCIA
'-----
```

**DIRECCION:**

SELECT CASE DIR

CASE 11,1

*' Adelante*

LOW DIR\_I

LOW DIR\_D

HPWM PWM\_I,VELOCIDAD1,500

HPWM PWM\_D,VELOCIDAD1+10,500

return

CASE 12

LOW DIR\_I

LOW DIR\_D

HPWM PWM\_I,VELOCIDAD1,500

HPWM PWM\_D,VELOCIDAD1+30,500

return

CASE 13

LOW DIR\_I

LOW DIR\_D

HPWM PWM\_I,VELOCIDAD1,500

HPWM PWM\_D,VELOCIDAD1+50,500

return

CASE 14

LOW DIR\_I

LOW DIR\_D

HPWM PWM\_I,VELOCIDAD1,500

HPWM PWM\_D,VELOCIDAD1+70,500

return

CASE 15

HIGH DIR\_I

LOW DIR\_D

HPWM PWM\_I,VELOCIDAD1+20,500

HPWM PWM\_D,VELOCIDAD1+20,500

return

CASE 2

LOW DIR\_I

LOW DIR\_D

HPWM PWM\_I,VELOCIDAD1+20,500

HPWM PWM\_D,VELOCIDAD1,500

return

CASE 3

LOW DIR\_I

LOW DIR\_D

HPWM PWM\_I,VELOCIDAD1+40,500

HPWM PWM\_D,VELOCIDAD1,500

return

CASE 4

LOW DIR\_I

LOW DIR\_D

HPWM PWM\_I,VELOCIDAD1+60,500

HPWM PWM\_D,VELOCIDAD1,500

return

CASE 5

LOW DIR\_I

HIGH DIR\_D

HPWM PWM\_I,VELOCIDAD1+20,500

HPWM PWM\_D,VELOCIDAD1+20,500

return

END SELECT

-----

*'DIRECTIVAS DE MOVIMIENTO BASICO*

-----

**ADELANTE:**

LOW DIR\_I  
LOW DIR\_D  
HPWM PWM\_I,90,800  
HPWM PWM\_D,100,800  
RETURN

**ATRAS:**

HIGH DIR\_I  
HIGH DIR\_D  
HPWM PWM\_I,90,500  
HPWM PWM\_D,100,500  
RETURN

**IZQUIERDA:**

LOW DIR\_I  
LOW DIR\_D  
HPWM PWM\_I,90,500  
HPWM PWM\_D,120,500  
RETURN

**DERECHA:**

LOW DIR\_I  
LOW DIR\_D  
HPWM PWM\_I,130,500  
HPWM PWM\_D,100,500  
RETURN

**PARAR:**

HIGH DIR\_I  
HIGH DIR\_D  
HPWM PWM\_I,0,500  
HPWM PWM\_D,0,500  
RETURN

-----  
'ESCANEO DE MAXSONAR  
-----

**MAX\_MAIN:**

```
    PORTA=%00000000          'Selección MAX_LI
    GOSUB MAX_SER
    GOSUB MAX_PROMEDIO
    MAX_LI=PROMEDIO
    PORTA=%00000001          'Selección MAX_CI
    GOSUB MAX_SER
    GOSUB MAX_PROMEDIO
    MAX_CI=PROMEDIO
    PORTA=%00000010          'Selección MAX_CD
    GOSUB MAX_SER
    GOSUB MAX_PROMEDIO
    MAX_CD=PROMEDIO
    PORTA=%00000011          'Selección MAX_LD
    GOSUB MAX_SER
    GOSUB MAX_PROMEDIO
    MAX_LD=PROMEDIO
    PORTA=%00001000
```

RETURN

-----  
'5 MEDICIONES X CADA SENSOR ULTRASONICO  
-----

**MAX\_SER:**

```
    FOR J=0 TO 4
        Serin2
        MAXSin,baudMAX,800,MAX_TIMEOUT,[wait("R"),GPS_RX[0],GPS_RX[1],GPS_RX[2]
    ]
        MAX_RES[J]=(GPS_RX[0]-48)*100+(GPS_RX[1]-48)*10+(GPS_RX[2]-48)
```

```
NEXT J
RETURN
```

-----  
'PROMEDIO DE LAS 5 MEDICIONES

**MAX\_PROMEDIO:**

```
CONT_ZERO=0
```

```
CONT_ONE=0
```

```
PROMEDIO=0
```

```
FOR j=0 TO 4
```

```
  IF MAX_RES[J]<10 THEN
```

```
    CONT_ZERO=CONT_ZERO+1
```

```
  ELSE
```

```
    IF MAX_RES[J]>150 THEN
```

```
      CONT_ONE=CONT_ONE+1
```

```
    ELSE
```

```
      PROMEDIO=PROMEDIO+MAX_RES[J]
```

```
    ENDIF
```

```
  ENDIF
```

```
NEXT J
```

```
IF CONT_ZERO>2 THEN
```

```
  PROMEDIO=6
```

```
ELSE
```

```
  IF CONT_ONE>2 THEN
```

```
    PROMEDIO=150
```

```
  ELSE
```

```
    PROMEDIO=PROMEDIO/(5-CONT_ZERO+CONT_ONE)
```

```
  ENDIF
```

```
ENDIF
```

```
RETURN
```

-----  
'MOSTRAR DATOS MAX  
-----

**MAX\_LCD:**

Lcdout \$FE, 1

LCDOUT \$FE, \$0F

Icdout " Ci:",DEC MAX\_CI," Cd:",DEC MAX\_CD

Lcdout \$FE, \$C0

Lcdout "Li:",DEC MAX\_LI," Ld:",DEC MAX\_LD

RETURN  
-----

'MANEJO DE TECLADO  
-----

**MAIN\_KEYPAD:**

SHIFT=%0

*'primero habilita shift*

Lcdout \$FE, 1

LCDOUT \$FE, \$0F

LCDOUT "DESTINO"

Lcdout \$FE, \$C0

LCDOUT "A ALCANZAR"

PAUSE 500

Lcdout \$FE, 1

LCDOUT \$FE, \$0E

LCDOUT \$FE, \$C0

'PETICIÓN COORDENADAS DESTINO

LCDOUT "LON:"

LCDOUT \$FE, 2

LCDOUT "LAT:"

GOSUB KEYPAD

RETURN

## KEYPAD:

GOSUB SKANKEYS

IF SHIFT=1 AND RESULT=1 THEN

    LCDOUT "N"

*'carácter N después de sumar 48*

ENDIF

IF SHIFT=1 AND RESULT=2 THEN

    LCDOUT \$FE, 2

*'cursor en la primera línea*

    LCDOUT \$FE, \$14

    LCDOUT \$FE, \$14

    LCDOUT \$FE, \$14

    LCDOUT \$FE, \$14

ENDIF

IF SHIFT=1 AND RESULT=3 THEN

    LCDOUT "E"

*'carácter e después de sumar 48*

ENDIF

IF SHIFT=1 AND RESULT=4 THEN

LCDOUT \$FE, \$10 *'cursor izquierda*

IF SHIFT=1 AND RESULT=6 THEN

LCDOUT \$FE, \$14 *'cursor derecha*

IF SHIFT=1 AND RESULT=7 THEN

    LCDOUT "S"

*'carácter S después de sumar 48*

ENDIF

IF SHIFT=1 AND RESULT=8 THEN

    LCDOUT \$FE, \$C0

*'cursor en la segunda línea*

    LCDOUT \$FE, \$14

    LCDOUT \$FE, \$14

    LCDOUT \$FE, \$14

    LCDOUT \$FE, \$14

ENDIF

IF SHIFT=1 AND RESULT=9 THEN

    LCDOUT "W"

*'carácter o después de sumar 48*

ENDIF

IF SHIFT=1 AND RESULT=10 THEN

    GOSUB READ\_coord

*'rutina para guardar coordenadas*

    RETURN

ENDIF

IF SHIFT=0 AND RESULT<10 THEN LCDOUT OLDRESULT+48

IF SHIFT=0 AND RESULT=10 THEN LCDOUT "."

PAUSE 200

GOTO KEYPAD

-----

'SCANEADO DE TECLAS

-----

**SKANKEYS:**

*'regresa la tecla en oldresul*

PORTB=%1000

*'1ra columna rb3=1*

GOSUB ROW

*'buscar 1ra fila*

IF FLAG=1 THEN FIRSTCOLUMN

*'valor de la tabla*

PORTB=%0100

*'2da columna rb2 = 1*

GOSUB ROW

IF FLAG=1 THEN SECONDCOLUMN

PORTB=%0010

*'3ra columna rb1 = 1*

GOSUB ROW

IF FLAG=1 THEN THIRDCOLUMN

Result=12

*'ninguna tecla seleccionada*

RETURN

-----

**FIRSTCOLUMN:**

LOOKUP KeyPress,[1,4,7,11],Result  
columna

*'carácter presionado primera*

OldResult = Result

IF OldResult=11 THEN

SHIFT=SHIFT ^ %00000001

IF SHIFT=%0 THEN

```
    LCDOUT $FE, $0E
ELSE
    LCDOUT $FE, $0F
ENDIF
ENDIF
RETURN
```

-----  
**SECONDCOLUMN:** *'carácter presionado primera columna*

```
    LOOKUP KeyPress,[2,5,8,0 ],Result
    OldResult = Result
RETURN
```

-----  
**THIRDCOLUMN:** *'carácter presionado primera columna*

```
    LOOKUP KeyPress,[3,6,9,10],Result
    OldResult = Result
RETURN
```

'ROWS-----

**ROW:**

```
KEYPRESS=0 'limpia keypress
    Flag=1 ' 1 si es presionada

if PORTB.4=1 then ExitRow ' sale si es presionada en i
KeyPress=KeyPress+1
    if PORTB.5=1 then ExitRow ' sale si es presionada en ii
    KeyPress=KeyPress+1
        if PORTB.6=1 then ExitRow ' sale si es presionada en iii
        KeyPress=KeyPress+1
            if PORTB.7=1 then ExitRow ' sale si es presionada en iv
            KeyPress=KeyPress+1
                Flag=0 ' tecla no presionada
```

ExitRow:

RETURN

-----  
'LEER COORDENADAS GPS Y ALMACENARLAS EN LA EEPROM  
-----

**READ\_COORD:**

```
LCDIN $84,[GPS_RX[11],GPS_RX[12],GPS_RX[13],GPS_RX[14]_
      ,GPS_RX[15],GPS_RX[16],GPS_RX[17],GPS_RX[18],_
      GPS_RX[19],GPS_RX[21]]           'lee latitud de LCD
GOSUB ASSCI_DEC
WRITE 20,GPS_RX[11]                   '
WRITE 21,GPS_RX[12]                   ' Latitud
WRITE 22,GPS_RX[13]                   '
WRITE 23,GPS_RX[14]                   '
WRITE 24,GPS_RX[16]                   '
WRITE 25,GPS_RX[17]                   '
WRITE 26,GPS_RX[18]                   '
WRITE 27,GPS_RX[19]                   '
WRITE 28,GPS_RX[21]+48                 '
LCDIN $C4,[GPS_RX[23],GPS_RX[24],GPS_RX[25],GPS_RX[26]_
      ,GPS_RX[27],GPS_RX[28],GPS_RX[29],GPS_RX[30],GPS_RX[31]_
      ,GPS_RX[32],GPS_RX[34]]         'lee longitud de LCD
GOSUB ASSCI_DEC
WRITE 30,GPS_RX[23]                   '
WRITE 31,GPS_RX[24]                   ' Longitud
WRITE 32,GPS_RX[25]                   '
WRITE 33,GPS_RX[26]                   '
WRITE 34,GPS_RX[27]                   '
WRITE 35,GPS_RX[29]                   '
WRITE 36,GPS_RX[30]                   '

```

```

WRITE 37,GPS_RX[31]
WRITE 38,GPS_RX[32]
WRITE 39,GPS_RX[34]+48
GOSUB BUFFER_PD
LGD=GP_LATD[2]*10+GP_LATD[3]
LMD=GP_LATD[4]*10+GP_LATD[5]
LSD=GP_LATD[6]*10+GP_LATD[7]
LSSD=GP_LATD[8]*10+GP_LATD[9]
GGD=GP_LOND[0]*100+GP_LOND[1]*10+GP_LOND[2]
GMD=GP_LOND[3]*10+GP_LOND[4]
GSD=GP_LOND[5]*10+GP_LOND[6]
GSSD=GP_LOND[7]*10+GP_LOND[8]
RETURN
'RECEPCION SERIAL SENTENCIA GGA
'-----
GPS_RX_GRGGA:
  Serin2 GPSin,baudGPS,2000,GPS_TIMEOUT,[wait("$GPGGA,"),GPS_RX[0],_
    GPS_RX[1],GPS_RX[2],GPS_RX[3],GPS_RX[4],GPS_RX[5],GPS_RX[6],_
    GPS_RX[7],GPS_RX[8],GPS_RX[9],GPS_RX[10],GPS_RX[11],GPS_RX[12]_
    ,GPS_RX[13],GPS_RX[14],GPS_RX[15],GPS_RX[16],GPS_RX[17],GPS_RX[1
8]_
    ,GPS_RX[19],GPS_RX[20],GPS_RX[21],GPS_RX[22],GPS_RX[23],GPS_RX[2
4]_
    ,GPS_RX[25],GPS_RX[26],GPS_RX[27],GPS_RX[28],GPS_RX[29],GPS_RX[3
0]_
    ,GPS_RX[31],GPS_RX[32],GPS_RX[33],GPS_RX[34],GPS_RX[35],GPS_RX[3
6]_
    ,GPS_RX[37],GPS_RX[38],GPS_RX[39]]
RETURN
'TIEMPOS DE ESPERA DE TRANSMISION SERIAL AGOTADOS
'-----

```

**GPS\_TIMEOUT:** *'Tiempo de espera comunicación GPS*

```
        GPS_FIX_TYPE = 0
    Lcdout $FE, 1          ' Limpia Display
    Lcdout $FE, 2
    LCDOUT "GPS_FIX=",GPS_FIX_TYPE
    GOSUB PARAR
    GOTO MAIN
```

**MAX\_TIMEOUT:** *'Tiempo de espera comunicación modulo sonar*

```
        lcdout $FE, 1
    Lcdout $FE, 2
    LCDOUT "MAX TIMEOUT!!!"
    GOSUB PARAR          ' Indica no MAX
    GOTO MAIN
```

**PIC\_TIMEOUT:** *'Tiempo de espera comunicación PIC*  
*18F452*

```
    Lcdout $FE, 1
    Lcdout $FE, 2
    LCDOUT "PIC SERIAL OUT!!"
    GOSUB PARAR
    serout2 P18out,baudP18,[dec BEARING,"OK"]
    Serin2 P18in,baudP18,500,PIC_TIMEOUT,[wait("RM"),GPS_RX[0]],
    goto MAIN          ' Indica no PIC
```

-----  
*'ASCII a DECIMAL*  
-----

**ASSCI\_DEC:**

```
    FOR i=0 TO 39
        GPS_RX[i]=GPS_RX[i]-48
    NEXT i
```

RETURN

-----  
'EXTRACCION COORDENADAS GPS  
-----

**EXTRAC\_COOR\_GPS:**

REPEAT

GOSUB GPS\_RX\_GRGGA

GOSUB BUFFER\_PS *'llenar el vector*

GOSUB ASSCI\_DEC

GPS\_HORAS = GPS\_RX[1] + (GPS\_RX[0] \* 10)

GPS\_MINUTOS = GPS\_RX[3] + (GPS\_RX[2] \* 10)

GPS\_SEGUNDOS = GPS\_RX[5] + (GPS\_RX[4] \* 10)

GPS\_LAT\_GRADOS = GPS\_RX[12] + (GPS\_RX[11] \* 10)

GPS\_LAT\_MINUTOS = GPS\_RX[14] + (GPS\_RX[13] \* 10)

GPS\_LAT\_SEGUNDOS = GPS\_RX[17] + (GPS\_RX[16] \* 10)

GPS\_LAT\_mSEGUNDOS = GPS\_RX[19] + (GPS\_RX[18] \* 10)

GPS\_LAT\_DIRECCION = GPS\_RX[21] + 48

GPS\_LON\_GRADOS = (GPS\_RX[23] \* 100)

GPS\_LON\_GRADOS = GPS\_LON\_GRADOS + (GPS\_RX[24] \* 10)

GPS\_LON\_GRADOS = GPS\_LON\_GRADOS + GPS\_RX[25]

GPS\_LON\_MINUTOS = GPS\_RX[27] + (GPS\_RX[26] \* 10)

GPS\_LON\_SEGUNDOS = GPS\_RX[30] + (GPS\_RX[29] \* 10)

GPS\_LON\_mSEGUNDOS = GPS\_RX[32] + (GPS\_RX[31] \* 10)

GPS\_LON\_DIRECCION = GPS\_RX[34] + 48

GPS\_FIX\_TYPE = GPS\_RX[36]-48

GPS\_SATS\_IN\_VIEW = (GPS\_RX[39]-48) + ((GPS\_RX[38]-48) \* 10)

until (GPS\_FIX\_TYPE<>5)

RETURN

-----  
'VECTOR DE COORDENADAS DEL DESTINO

---

**BUFFER\_PD:**

```
GP_LATD[0]="P"  
GP_LATD[1]="D"  
READ 20,GP_LATD[2]  
READ 21,GP_LATD[3]  
READ 22,GP_LATD[4]  
READ 23,GP_LATD[5]  
READ 24,GP_LATD[6]  
READ 25,GP_LATD[7]  
READ 26,GP_LATD[8]  
READ 27,GP_LATD[9]  
READ 28,GP_LATD[10]  
GP_LATd[11]="O"  
GP_LATd[12]="K"  
READ 30,GP_LOND[0]  
READ 31,GP_LOND[1]  
READ 32,GP_LOND[2]  
READ 33,GP_LOND[3]  
READ 34,GP_LOND[4]  
READ 35,GP_LOND[5]  
READ 36,GP_LOND[6]  
READ 37,GP_LOND[7]  
READ 38,GP_LOND[8]  
READ 39,GP_LOND[9]  
GP_LOND[10]="O"  
GP_LOND[11]="K"  
RETURN
```

---

*'VECTOR DE COORDENADAS DEL ORIGEN*

---

**BUFFER\_PS:**

GP\_LATS[0]="P"  
GP\_LATS[1]="S"  
GP\_LATS[2]=GPS\_RX[11]  
GP\_LATS[3]=GPS\_RX[12]  
GP\_LATS[4]=GPS\_RX[13]  
GP\_LATS[5]=GPS\_RX[14]  
GP\_LATS[6]=GPS\_RX[16]  
GP\_LATS[7]=GPS\_RX[17]  
GP\_LATS[8]=GPS\_RX[18]  
GP\_LATS[9]=GPS\_RX[19]  
GP\_LATS[10]=GPS\_RX[21]  
GP\_LATS[11]="O"  
GP\_LATS[12]="K"  
GP\_LONS[0]=GPS\_RX[23]  
GP\_LONS[1]=GPS\_RX[24]  
GP\_LONS[2]=GPS\_RX[25]  
GP\_LONS[3]=GPS\_RX[26]  
GP\_LONS[4]=GPS\_RX[27]  
GP\_LONS[5]=GPS\_RX[29]  
GP\_LONS[6]=GPS\_RX[30]  
GP\_LONS[7]=GPS\_RX[31]  
GP\_LONS[8]=GPS\_RX[32]  
GP\_LONS[9]=GPS\_RX[34]  
GP\_LONS[10]="O"  
GP\_LONS[11]="K"

*FINAL*

---

LGS=(GP\_LATS[2]-48)\*10+(GP\_LATS[3]-48)

```
LMS=(GP_LATS[4]-48)*10+(GP_LATS[5]-48)
LSS=(GP_LATS[6]-48)*10+(GP_LATS[7]-48)
LSSS=(GP_LATS[8]-48)*10+(GP_LATS[9]-48)
GGS=(GP_LONS[0]-48)*100+(GP_LONS[1]-48)*10+(GP_LONS[2]-48)
GMS=(GP_LONS[3]-48)*10+(GP_LONS[4]-48)
GSS=(GP_LONS[5]-48)*10+(GP_LONS[6]-48)
GSSS=(GP_LONS[7]-48)*10+(GP_LONS[8]-48)
```

```
RETURN
```

```
-----
```

```
' MOSTRAR COORDENADAS
```

```
-----
```

```
LCD_GPS_COOR:
```

```
LCDOUT $FE, 1           ' Limpiar display
LCDOUT $FE, 2           ' Comienzo de la primera línea
LCDOUT "LAT:",GP_LATS[2],GP_LATS[3],GP_LATS[4],GP_LATS[5],".",_
                    GP_LATS[6],GP_LATS[7],GP_LATS[8],GP_LATS[9],_
                    GP_LATS[10],DEC GPS_SATS_IN_VIEW
LCDOUT $FE, $C0         ' Comienzo de la segunda línea
LCDOUT "LON:",GP_LONS[0],GP_LONS[1],GP_LONS[2],GP_LONS[3]_
                    ,GP_LONS[4],".",GP_LONS[5],GP_LONS[6],GP_LONS[7]_
                    ,GP_LONS[8],GP_LONS[9],DEC GPS_FIX_TYPE
```

```
RETURN
```

```
-----
```

## ANEXO D: CODIGO FUENTE CAMPO DE FUERZAS VIRTUALES

### program VFF

const pi2 = pi\*2

const pim = pi/2

const Ol = 40\*pi/180 *'ángulo sensores laterales 40*

const Oc = 13\*pi/180 *'ángulo sensores centrales 13*

const Otp = 0.82\*pi *'ángulo modo sigue pared*

dim LAs,LGs,LAd,LGd as float

dim O\_t,O\_i,O\_r,O\_v,oaux,Frumbo as float *'ángulo de objetivo, instantáneo y de evasión*

dim dirN\_s,dirE\_s,dirN\_d,dirE\_d,VI as integer *'Dirección geográfica*

dim txt,out,Velocidad as string[10]

dim txt1 as string[12]

dim PS,PL as string[22] *'vectores de almacenamiento punto de salida y de destino*

dim M\_li,M\_ci,M\_ld,M\_cd as float *'distancias medidas por lo sensores*

dim Motion,dir1 as char *'dirección de movimiento y grado de giro*

**sub procedure latlong**(dim byref LA,LG as float,dim byref dirNS,dirEW as INTEGER,dim byref rx as string[21],dim SD as char)

*'conversión de caracteres a grados y radianes de las coordenadas geográficas*

dim aux,res as float

res=0

if rx[1]=SD then *'*

Aux=(((rx[8]-48)\*10)+(rx[9]-48))/100 +(((rx[6]-48)\*10)+(rx[7]-48))

res=(res+Aux)/60

Aux=(((rx[4]-48)\*10)+(rx[5]-48))

res=(res+Aux)/60

Aux=(((rx[2]-48)\*10)+(rx[3]-48))

LA=(res+Aux)\*pi/180

```

.....

res=0

Aux=((((rx[18]-48)*10)+(rx[19]-48))/100)+(((rx[16]-48)*10)+(rx[17]-48))

res=100*(res+Aux)/6

Aux=1000*(((rx[14]-48)*10)+(rx[15]-48))

res=(res+Aux)/60

LG=res

IF rx[10]="N" THEN
  dirNS=1
ELSE
  dirNS=-1
END IF

IF rx[20]="W" THEN
  dirEW=1
ELSE
  dirEW=-1
END IF

end if

end sub

sub function rumbo(dim Lts1,Los1,ltd1,lod1 as float, dim La_s,Lg_s,La_d,Lg_d as
float) as float

  dim L,g,Ap,Lm,ri,rvi,a,b as float

  dim rl,rg as integer

  'producto latitud longitud por símbolo N(+),S(-),E(-),W(+)
  '-----

  'if La_s=La_d then

    IF Ltd1>=lts1 then

      L=Ltd1-Lts1

      if la_s=-1 then

        rl=-1

      else

```

```
        rl=1
    end if
else
    L=Lts1-Ltd1
    if la_s=-1 then
        rl=1
    else
        rl=-1
    end if
end if
'else
' L=Lts1+Ltd1
' rl=la_d
'end if
' Lg_s=Lg_d then
    IF Lod1>=los1 then
        g=Lod1-Los1
        if lg_s=-1 then
            rg=-1
        else
            rg=1
        end if
    else
        g=Los1-Lod1
        if lg_s=-1 then
            rg=1
        else
            rg=-1
        end if
    end if
end if
```

```

'else
  'g=Los1+Lod1
  'rl=lg_d
'end if

'calculo del rumbo, metodo del circulo maximo
b=g*pi/180000 'b
g=b
Lm=(Lts1+Ltd1)/2
Ap=g*cos(Lm)
ri=atan(Ap/L)
if (rl=1) and (rg=1) then
  rvi=pi2-ri
end if
if (rl=1) and (rg=-1) then
  rvi=ri
end if
if (rl=-1) and (rg=1) then
  rvi=pi+ri
end if
if (rl=-1) and (rg=-1) then
  rvi=pi-ri
end if

'   'eje de referencia y +
'-----

'movimiento en el mismo polo o paralelo
if (L=0) THEN
  if rg=1 then
    rvi=3*pim
  else
    rvi=pim
  
```

```

    end if
else
    if (g=0) THEN
        if rl=1 then
            rvi=0
        else
            rvi=pi
        end if
    end if
end if

'-----
result = rvi*1800/pi
'-----

end sub

sub procedure rumbotx(dim byref r as string[10],dim byref artx as string[7])
    dim i,j,k as byte
    k=5
    i=Strchr(r, ".")
    artx="RM0000"
    for j=(i-1) to 0 step -1
        artx[k]=r[j]
        k=k-1
    next j

end sub

sub procedure Frepulsiva(dim byref Qr,ModFr,FrX,Fry as float)
    dim x,y,Oli,Oci,Ocd,Old,O_o,MLip,MCip,Mcdp,Mldp as float
    Frx=0
    Fry=0

```

$$O_o = \text{pim} - O_i$$

*'ángulo instantáneo en el eje x+*

$$O_{li} = O_o + O_l$$

$$O_{ci} = O_o + O_c$$

$$O_{cd} = O_o - O_c$$

$$O_{ld} = O_o - O_l$$

$$M_{lip} = (6250/M_{Li}) - 42$$

$$M_{cip} = (6250/M_{ci}) - 42$$

$$M_{cdp} = (6250/M_{cd}) - 42$$

$$M_{ldp} = (6250/M_{Ld}) - 42$$

$$x = -M_{lip} \cdot \cos(O_{li})$$

$$y = -M_{lip} \cdot \sin(O_{li})$$

$$F_x = F_x + x$$

$$F_y = F_y + y$$

$$x = -M_{cip} \cdot \cos(O_{ci})$$

$$y = -M_{cip} \cdot \sin(O_{ci})$$

$$F_x = F_x + x$$

$$F_y = F_y + y$$

$$x = -M_{ldp} \cdot \cos(O_{ld})$$

$$y = -M_{ldp} \cdot \sin(O_{ld})$$

$$F_x = F_x + x$$

$$F_y = F_y + y$$

$$x = -M_{cdp} \cdot \cos(O_{cd})$$

$$y = -M_{cdp} \cdot \sin(O_{cd})$$

$$F_x = F_x + x$$

$$F_y = F_y + y$$

$$\text{ModFr} = \sqrt{\text{Pow}(F_x, 2) + \text{Pow}(F_y, 2)}$$

$$Q_r = \text{Atan2}(F_x, F_y)$$

*'ángulo en el eje x+*

$$Q_r = \text{pim} - Q_r$$

if  $Q_r < 0$  then

$$Q_r = \text{pi} + Q_r$$

```

        end if                                     'ángulo desde el eje y+ norte
    '-----
end sub

sub function Fuerza_virtual() as float
    dim Ftx,Fty,FAx,FAy,ModFr,Frx,Fry,Ow,Ao,phi as float
        Frepulsiva(Or,ModFr,Frx,Fry)
        FAx=340*cos(pim-Ot)
        FAy=340*sin(pim-Ot)
    'Sumatoria Fa+Fr fuerza resultante
        Ftx=FAx+Frx
        Fty=FAy+Fry
    'ángulo q debe recorrer para evitar obstáculos y alcanzar destino
    '-----
    'mod_f=Sqrt(Pow(Sx,2)+Pow(Sy,2))
        Ov=Atan2(Ftx,Fty)
        Ov=pim-Ov
        if Ov<0 then
            Ov=pi2+Ov
        end if
    'Control Velocidad en areas
    if (ModFr>=0) and (ModFr<10) then
        Velocidad="120"+
    else
        if (MLi<80) or (MLd<80) or (Mci<80) or (Mcd<80) then
            if (MLi<50) or (MLd<50) or (Mci<50) or (Mcd<50) then
                Velocidad="070"
            else
                Velocidad="090"
            end if
        else
    end if
end sub

```

```

                                velocidad="100"
                                if (Fabs(O_r-O_t)>=2.967) and (Fabs(O_r-O_t)<=3.3) then '(170--
190)                                O_v=O_v+0.785    '(45)
                                end if
                                end if
                                end if

```

```

-----
Ao=Fabs(O_t-O_i)
' Algoritmo sigue pared
-----

if (Ao>pim) and (Ao<pi) then '90-180 ' 75-285 (Ao>1.31) and (Ao<4.97)
    Lcd_Out(1, 1,"sigue pared ")
    'IF dir1="1" THEN
        O_v=O_i-pim
    'ELSE
        ' O_v=O_i+pim
    ' END IF
    'if (O_t<O_i) then
        ' Ov=O_r-Otp
    ' else
        ' Ov=O_r+Otp
    ' end if
' else
    ' Ov=pim-O_t 'eje de referencia x +
    ' if (Fabs(O_r-Ot)>=2.967) and (Fabs(O_r-Ot)<=3.3) then '(170--190)
        ' Ov=Ov+0.785    '(45)
    ' end if
end if

```

```

-----
' Control de dirección

```

```

if O_i>O_v then
  phi=O_i-O_v
  if (phi>pi) then
    phi=(pi2-phi)
    dir1="0"
  else
    dir1="1"
  end if
else
  phi=O_v-O_i
  if (phi>pi) then
    phi=(pi2-phi)
    dir1="1"
  else
    phi=phi
    dir1="0"
  end if
end if

```

-----

*'Control de giro (angular)*

-----

```

Motion="0"
if (phi>=0) and (phi<0.1745) then Motion="1" end if ' 0-10()' 0-5(0.087)
if (phi>=0.1745) and (phi<0.6981) then Motion="2" end if ' 10-40
if (phi>=0.6981) and (phi<1.4835) then Motion="3" end if ' 40-85
if (phi>=1.4835) and (phi<2.5307) then Motion="4" end if ' 85-145
if (phi>=2.5307) and (phi<=pi) then Motion="5" end if ' 145-180

```

-----

```

    result=O_v*1800/pi
end sub

sub function LCD_angulos() as float
    'Visualizacion de los angulos calculados
    O_i=O_i*180/pi
    O_t=O_t*180/pi
    O_r=O_r*180/pi
    O_v=O_v*180/pi
    FloatToStr(O_i,out)
    Lcd_Cmd(Lcd_Clear)
    Lcd_Out(1, 1,"O")
    Lcd_Out(1, 2,out)
    FloatToStr(O_r,out)
    Lcd_Out(1, 10,"R")
    Lcd_Out(1, 11,out)
    FloatToStr(O_t,out)
    Lcd_Out(2, 1,"T")
    Lcd_Out(2, 2,out)
    FloatToStr(O_v,out)
    Lcd_Out(2, 10,"V")
    Lcd_Out(2, 11,out)
    result=0
end sub

```

**main:**

```

    Usart_Init(9600) 'Configuración comunicación serial
    ADCON1 = $0F
    Lcd_Config(PORTd,7,6,5,4,PORTd,1,2,3)
    lcd_Out(1,1,"VIRTUAL FORCE")
    LCD_OUT(2,1,"ESPERANDO!")

```



```
    rumbotx(txt1,txt)
    txt[6]=dir1
    txt[7]=Motion
    txt[8]=Velocidad[0]
    txt[9]=Velocidad[1]
    txt[10]=Velocidad[2]
    Usart_Write_text(txt)           'Transmisión de resultado
end if
wend
end.
```

**ANEXO E: PRUEBA DEL ROBOT (FOTOS Y VIDEOS)**

**Ver el Disco Adjunto**

**Ubicación /Pruebas**

## ANEXO F: MANUAL DE USUARIO

### Generalidades del sistema

Nav Sat es un sistema robótico desarrollado para detectar obstáculos en su camino cuando se encuentra dirigiéndose hacia las coordenadas que el usuario le haya indicado al inicio de su ejecución. Los sensores ultrasónicos reconocerán objetos en su frente siempre que cuenten con características de superficie adecuadas y de ubicación y altura, el módulo GPS cuenta con los estándares para detectar tantos satélites como le sea posible en la mayoría de situaciones de clima. La interfaz de usuario es muy intuitiva, se tiene un botón principal para la activación del sistema, posee un teclado básico de 12 teclas y un display principal donde se podrán verificar los ingresos, se tiene además teclas especiales que activan funciones secundarias de las otras teclas así como también modos de visualización de datos en el display.

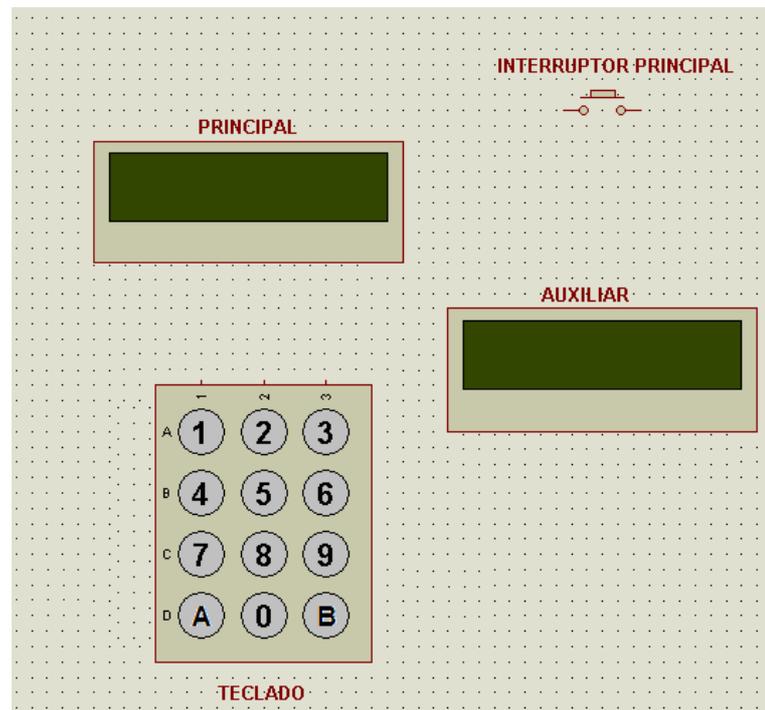


Figura1. Tablero de controles

Cuando la ejecución se inicia el robot decidirá su propio comportamiento en base lo que perciba de su entorno y procurará llevar a término la orden del usuario. Dado el factor de error ingresado por el sistema GPS se toma en

cuenta que el objetivo a alcanzar se encuentra dentro de un área de 2 metros de radio.

## **Funcionamiento del sistema**

### **1. Encendido del equipo**

Un botón del tipo pulsador en el tablero principal conecta todo el equipo a la fuente, habilitando así los dispositivos de entrada / salida de datos de usuario, para que pueda monitorear si todos los sensores se encuentran en línea o para proceder a ejecutar la navegación del móvil por el entorno.

### **2. El teclado**

La única interfaz humana de entrada de datos en este sistema es el teclado incorporado en el tablero principal, dotado de 12 teclas, 10 de ellas numeradas del 0 al 9 y las restantes señaladas con las primeras letras del alfabeto, utilizadas para ejecutar funciones secundarias.

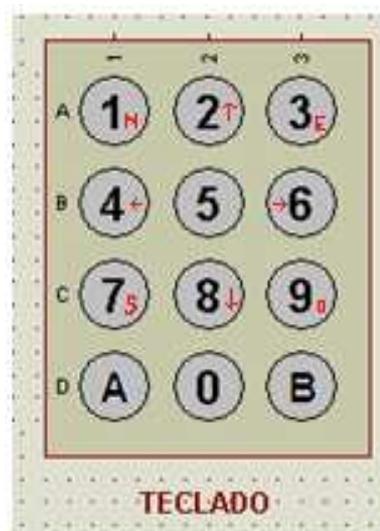


Figura 2. El teclado

## **Funciones al iniciar el sistema**

A través de una pantalla de bienvenida, el sistema se encuentra a la espera de alguna orden que llegue por el teclado en las siguientes teclas (ver apartado 5):

- **Tecla 1.** Monitoreo de coordenadas GPS
- **Tecla 4.** Monitoreo de brújula electrónica
- **Tecla 7.** Monitoreo del sonar ultrasónico
- **Tecla A.** Cancelación de monitoreo de sensores e ingreso al modo de entrada de datos

### **Funciones secundarias**

Estas funciones se habilitan cuando el sistema se encuentra en modo de entrada de datos:

- **Tecla A.** Esta tecla acciona el teclado alfabético para insertar las letras correspondientes a los puntos cardinales de las coordenadas, también acciona el cursor para realizar correcciones de entrada.
- **Tecla B.** Esta tecla tiene asociado el punto decimal pero combinada con la tecla A se convierte en el Enter que inicia la ejecución.
- **Tecla 4 y 6.** Funciones de cursor, izquierdo y derecho respectivamente
- **Tecla 1, 3, 7 y 9.** Letras para ingresar el punto cardinal, norte para el 1, sur para el 7, este para el 3 y oeste para el 9.

### **3. Ingreso de datos**

El formato para el ingreso de los datos debe tener una estructura definida, pues de lo contrario no se podrán utilizar y el software no podrá realizar su trabajo.

- Ingresar la latitud, seguidas de una letra que indicará el punto cardinal de la posición de llegada, utilizando tecla A mas tecla B, se ingresa un Enter para pasar a la siguiente línea.

- Ingresar la longitud, cuando ambas coordenadas estén correctamente ingresadas, se podrá enviar la orden de ejecución para que el robot inicie la búsqueda del objetivo presionando la combinación tecla A mas tecla B nuevamente.

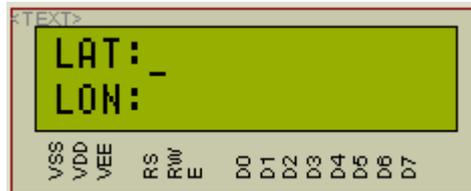


Figura 3. Ingreso de datos

### **Corrección de datos mal ingresados**

Utilizando la tecla A combinada con las teclas 4 y 6 que son el cursor izquierdo y derecho respectivamente, el cursor cambia de línea baja intermitente a cursor lleno intermitente, se podrá mover entre los caracteres de la línea, llegar al carácter que se desea corregir y sustituirlo por el correcto.

### **4. Ejecución**

Presionar la tecla A mas la tecla B para ingresar un Enter luego de haber ingresado las coordenadas de llegada, hará que el robot inicie la ejecución del programa de navegación.

### **5. Interpretación de los datos en el display**

El tablero tiene dos display, el de interés para el usuario es el display principal, muestra los siguientes datos ( ver apartado 2):

- Pantalla de ingreso de datos. El teclado y el display se habilitan para ingresar los datos de las coordenadas de llegada.

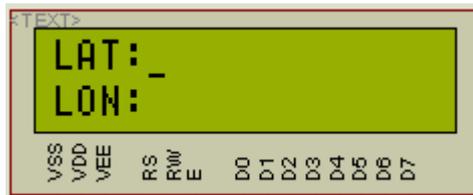


Figura 4. Ingreso de datos

- Pantalla de monitoreo sensores ultrasónicos. Se muestra la medida numérica que los sensores ultrasónicos están detectando al momento.

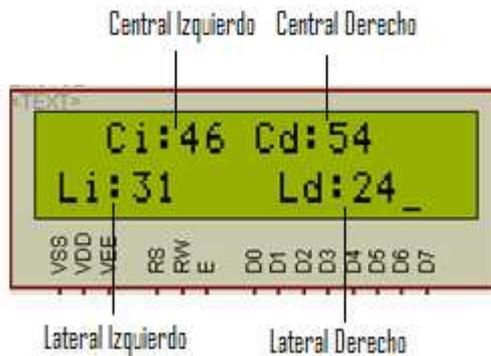


Figura 5. Monitoreo sensores ultrasonido

- Pantalla de monitoreo GPS. Se muestran las coordenadas del punto actual, tipo de fix y cantidad de satélites visibles.

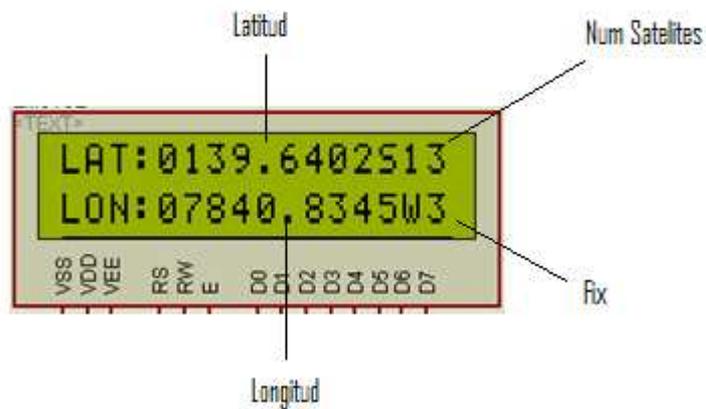


Figura 6. Monitoreo receptor GPS

- Pantalla de monitoreo brújula. Se muestra el ángulo detectado respecto al norte terrestre



Figura 7. Monitoreo brújula electrónica

## 6. Interrupción de la ejecución

En cualquier situación donde sea necesario interrumpir la ejecución del programa, tenemos dos opciones, la primera es apagar el sistema completamente con el pulsador de encendido apagado y la segunda es ordenar una interrupción presionando la tecla 1, logrando así que el sistema operativo vuelva al estado de espera de datos para reiniciar el procedimiento.

## 7. Uso del segundo display

Este display tiene como fin el monitoreo del micro controlador de cálculos matemáticos para poder controlar que el procesamiento se realice de forma constante, y conociendo sus resultados descartar ciertas etapas en caso de posibles fallas y errores de comunicación.

## 8. Mantenimiento

El robot móvil no reviste de mayores atenciones, salvo la batería que necesita ser recargada después de cada sesión de navegación. El módulo de control de los motores DC está montado al frente del robot y está provisto de un refrigerador pasivo que puede servir para determinar, según su temperatura, si esta etapa está funcionando correctamente.

Otra etapa que necesita de atención es la brújula electrónica ya que para su óptimo funcionamiento necesita estar en posición horizontal, situación que se facilita porque la brújula está montada en un soporte omni direccional y puede ser ajustada en todo momento.

## **9. Recomendaciones**

- Recargar la batería después de cada sesión de navegación.
- Utilizar el sistema en modo monitoreo como receptor GPS para determinar las posiciones iniciales y finales.
- Evitar terrenos accidentados ya que la tracción no se diseñó para tales entornos.
- Revisar el enfriador del controlador de los motores para monitorear si no hay sobrecalentamiento o cortos circuitos.
- Ajustar antes de cualquier prueba, la posición de la brújula para evitar cálculos erróneos.

## BIBLIOGRAFIA

1. ALEXANDER, J. Loxodromes: A Rhumb Way to Go. Cleveland: Case Western Reserve University, 2004. 77(5): 349-355.  
<http://www.cwru.edu/artsci/math/alexander/>  
2008-12
2. BETKE, K. The NMEA 0183 Protocol [en línea]. Mayo 2000. pp. 28.  
<http://www.scribd.com/doc/6365285/The-NMEA-0183-Protocol>  
2008-12
3. BORENSTEIN, J. ; KOREN, Y. Real-time Obstacle Avoidance for Fast Mobile Robots. IEEE Transactions on Systems, Man, and Cybernetics. [s.l.]. 19(5): 1179-1187. Sept/Oct. 1989.  
<http://www-personal.umich.edu/~johannb/>  
2008-12
4. CAÑAS, J. M. Navegación Global de un Robot usando el Método del Gradiente. Ing. en Informática de Sistemas. España. Universidad Rey Juan Carlos. Escuela Superior de Ciencias Experimentales y Tecnología, 2005. pp. 40.

5. CARVAJAL, J. H. Robótica: Aproximación al Diseño Mecatrónico.  
Colombia: Universidad de Pamplona, 2003. pp. 9-37.
6. CHINEA, C. S. Posicionándonos en el Globo Terrestre [en línea].  
<http://personales.ya.com/casanchi/ast/latierra.htm>  
2008-12
7. CHUNG, E. RD. SPK- GPS – GS405. Taipei: SPK ELECTRONICS CO.,  
LTD., September, 2006. 10F(510):1-14.
8. CHUNG YIN, K. H. Development of a GPS-based System for Monitoring  
Driver Performance. Thesis Electronic Engineer. U.S.A. University of  
Auckland. Department of Electrical and Electronic Engineering, 2003.  
pp. 14-21.
9. European Organization for the Safety of Air Navigation (EUROCONTROL).  
WGS 84 IMPLEMENTATION MANUAL. Brussels-Belgium: Institute of  
Geodesy and Navigation. University FAF Munich, Germany. Version 2.4,  
Febrero 12, 1998. pp. 13-26.
10. FERNANDEZ, I. A. Localizaciones Geográficas. Palencia. Universidad de  
Valladolid. pp. 3-11.  
[http://www.cartesia.org/data/apuntes/cartografia/ cartografia-  
geograficas.pdf](http://www.cartesia.org/data/apuntes/cartografia/cartografia-geograficas.pdf)  
2008-12
11. GONZÁLEZ, A. M. Iniciación a la Orientación [en línea].  
<http://www.escolar.com/avanzado/geografia008.htm>  
2008-12

12. GONZÁLEZ, M. A. Curso de Robótica. III Congreso Internacional Electrónica Y Tecnologías Avanzadas, 2003. Pamplona: Universidad de Pamplona, 2003. pp. 442.
13. HOGAN, J. E.; SCHMITT, P. R.; WAYNE J. Ultrasonic Sensor Model and Configuration Simulations for Mobile Robot Navigation in Narrow Aisles. Georgia: Woodruff School of Mechanical Engineering, diciembre 9, 1994. pp. 23.
14. HUMPHREYS, M.J.; HAMMERTON, C.J.; BROWN, D. Power Semiconductor Applications. Philips Semiconductors, 1994. pp. 283-300. <http://www.nxp.com>.  
2008-10
15. MARTÍN, B.; BONO, A. Enseñanza de una Metodología para la Programación de Microcontroladores en el marco de la Titulación de Electrónica Industrial. España: Universidad de Zaragoza, 2003. pp. 1-8.
16. MASSA, D. P. Choosing an Ultrasonic Sensor for Proximity or Distance Measurement, U.S.A, 1999.  
<http://www.sensorsmag.com/sensors/author/authorInfo.jsp?id=32343>  
2008-06
17. MAXBOTIX® INC. Frequently Asked Questions for the MaxSonar®-EZ1™ & the LV-MaxSonar®-EZ1™ [en línea].  
<http://www.maxbotix.com>  
2008-06
18. MAXBOTIX® INC. LV-MaxSonar®-EZ1™ Data Sheet. U.S.A. 2007. v3.0c.  
pp. 2. [www.maxbotix.com](http://www.maxbotix.com)  
2008-06

19. MICROCHIP. PIC16F877A Data Sheet. U.S.A.: Microchip Technology Incorporated, 2003. pp. 234.  
<http://ww1.microchip.com/downloads/en/DeviceDoc/39582b.pdf>  
2008-06
20. MICROCHIP. PIC18F452 Data Sheet. U.S.A.: Microchip Technology Incorporated, 2007. pp. 325.  
<http://ww1.microchip.com/downloads/en/DeviceDoc/39564c.pdf>  
2008-06
21. National Marine Electronics Association (NMEA). NMEA 2000® Network [en línea]. U.S.A. June 2005.  
[www.nmea.org](http://www.nmea.org)  
2008-12
22. OHYA, A.; OHNO, T.; YUTA, S. Obstacle Detectability of Ultrasonic Ranging System and Sonar Map Understanding. Japón: University of Tsukuba. Intelligent Robot Laboratory Institute of Information Sciences and Electronics, s.f . pp. 9.
23. PÉREZ, D. Sensores de distancia por Ultrasonidos [en línea].  
<http://alcabot.org/seminario2006/Trabajos/DiegoPerezDeDiego.pdf>  
2008-12
24. POZO-RUZ, A.; et al. Sistema de Posicionamiento Global (GPS): descripción, análisis de errores, aplicaciones y futuro. Arganda-Madrid: Instituto de Automática Industrial, 2000. p. 9.  
<http://www.iai.csic.es/users/gpa/postscript/Pozo-Ruz00a.pdf>  
2008-10

25. SALAZAR, D. J. Navegación Aérea, Cartografía y Cosmografía. España:  
Escuela Politécnica Superior de Castelldefels, 2008, Versión 6. pp. 3-7.
26. SANDIN, P. E. Robot Mechanisms and Mechanical Devices Illustrated.  
U.S.A. McGraw-Hil, 2003. pp. 69-155.
27. SIEGWART, R. Introduction to Autonomous Mobile Robots A Bradford  
Book. London-England: The MIT Press, 2004. pp. 104-108, 272-300
28. SOUTHWINGS. Loxodrómicas y Navegación por Estima. España:  
Southwings-CICERON, 2007, Versión 3. pp. 14.  
<http://www.southwings.net/>  
2008-10
29. SUPER ROBOTICA. Sensor Brújula Digital CMPS03 [en línea]. España:  
INTPLUS, 2008.  
<http://www.superrobotica.com>  
2008-06
30. TAURASO, C. Localizador GPS. Barcelona. In Electronics, 2005. pp. 1-4.