



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO

CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TEMA:

**DISEÑO E IMPLEMENTACIÓN A ESCALA DE UN ROBOT ACUÁTICO
RADIOCONTROLADO MEDIANTE COMUNICACIÓN INALÁMBRICA**

Previa la obtención del Título

INGENIERO EN TELECOMUNICACIONES

ELABORADO POR:

José Manuel Aguilar Jaramillo

Guayaquil, 02 de Mayo del 2014



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**FACULTAD TECNICA PARA EL DESARROLLO
CARRERA DE INGENIERIA EN TELECOMUNICACIONES CON MENSION
EN GESTION EMPRESARIAL.**

CERTIFICACIÓN

Certifico que el presente trabajo fue realizado en su totalidad por el Sr.
José Manuel Aguilar Jaramillo como requerimiento parcial para la obtención
del título de INGENIERO EN TELECOMUNICACIONES CON MENSION EN
GESTION EMPRESARIAL.

Guayaquil, 02 de Mayo del 2014

DIRECTOR

Msc. Edwin Palacios Meléndez

REVISADO POR

Msc. Bayardo Bohórquez Escobar
Revisor Metodológico

Ing. Carlos Romero Rosero
Revisor de Contenido

Msc. Armando Heras Sánchez
Director de Carrera



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**FACULTAD TECNICA PARA EL DESARROLLO
CARRERA EN INGENIERIA EN TELECOMUNICACIONES CON MENSION
EN GESTION EMPRESARIAL.**

DECLARACIÓN DE RESPONSABILIDAD

JOSÉ MANUEL AGUILAR JARAMILLO

DECLARÓ QUE:

El proyecto de tesis denominado **“Diseño e Implementación a escala de un Robot Acuático Radiocontrolado mediante Comunicación Inalámbrica”** ha sido desarrollado con base a una investigación exhaustiva, respetando derechos intelectuales de terceros conforme las citas que constan al pie de las páginas correspondientes, cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es de mi autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del proyecto de grado en mención.

Guayaquil, 02 de Mayo del 2014

EL AUTOR

JOSÉ MANUEL AGUILAR JARAMILLO



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**FACULTAD TECNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES CON MENSION
EN GESTION EMPRESARIAL**

AUTORIZACIÓN

Yo, JOSÉ MANUEL AGUILAR JARAMILLO

Autorizó a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del proyecto titulado: **“Diseño e Implementación a escala de un Robot Acuático Radiocontrolado mediante Comunicación Inalámbrica”**, cuyo contenido, ideas y criterios es de mi exclusiva responsabilidad y autoría.

Guayaquil, 02 de Mayo del 2014

EL AUTOR

JOSÉ MANUEL AGUILAR JARAMILLO

DEDICATORIA

Mi tesis la dedico con todo mi amor y cariño:

A ti Dios que me distes la oportunidad de vivir y de regalarme una familia maravillosa, Con mucho cariño a mis Padres, que a pesar de todas las dificultades que vivimos siempre han estado conmigo en apoyando en mis estudios dando aliento y ánimos en todo momento.

Gracias papi y mami, por darme la mejor herencia que un padre de familia le puede dar a sus hijos, una carrera para mi futuro y

Por creer en mí, por todo esto les agradezco de todo corazón que estén

Conmigo a mi lado, solamente les estoy devolviendo lo que ustedes

Me dieron al principio.

A mi hermana Estefany y su esposo, y a mi querida sobrina "Dulce María" a quien aprecio y quiero mucho.

EL AUTOR

JOSÉ MANUEL AGUILAR JARAMILLO

AGRADECIMIENTO

A mi director de Tesis: MsC. Edwin Fernando Palacios Meléndez. Por su asesoramiento y su predisposición permanente en aclarar mis dudas y brindarme sus sugerencias en la redacción de la tesis.

A todos los profesores que me transmitieron sus conocimientos, vivencias. Los cual me ha servido a lo largo de toda mi carrera.

Al Ing. Carlos Romero Rosero ya que por circunstancias de la vida me puso en una situación complicada y él siempre estuvo pendiente del estado de mi padre y de mi situación académica, dando ánimos y palabras de aliento, para no dejarse llevar de la situación y fracasar.

EL AUTOR

JOSÉ MANUEL AGUILAR JARAMILLO

Índice General

Índice de Figuras	IX
Índice de Tablas	XI
CAPÍTULO 1: GENERALIDADES DEL TRABAJO DE TITULACIÓN	1
1.1. Introducción.	1
1.2. Antecedentes.	1
1.3. Justificación del Problema.	2
1.4. Definición del Problema.	2
1.5. Objetivos del Problema de Investigación.	2
1.5.1. Objetivo General.	2
1.5.2. Objetivos Específicos.	2
1.6. Idea a Defender.	3
1.7. Metodología de Investigación.	3
CAPÍTULO 2: Estado del Arte de Sistema Microcontroladores PIC en Robótica.	4
2.1. ¿Qué es un Microcontrolador?	4
2.2. ¿Por qué usar un Microcontrolador?	5
2.3. El Microcontrolador PIC	6
2.3.1. Memoria de Programa.	7
2.3.2. Memoria de Datos.	10
2.3.3. Líneas de E/S para módulos periféricos.	13
2.4. Programación MikroBasic para PIC's.	19
2.5. Características de programación en Basic.	20
2.5.1. Identificadores.	21
2.5.2. Etiquetas.	22
2.5.3. Constantes.	23

2.5.4. Variables.....	24
2.5.5. Símbolos.....	25
CAPÍTULO 3: Comunicaciones Inalámbricas.....	26
3.1. Introducción a Redes Inalámbricas.....	26
3.2. Propagación Inalámbrica.....	29
3.3. Estándares de Redes Inalámbricas 802.11.....	31
3.4. Estándar Home RF.....	33
3.5. Estándar HIPERLAN/1 e HIPERLAN/2.....	34
3.6. Redes Inalámbricas con tecnología WiFi.....	35
CAPÍTULO 4: DESARROLLO EXPERIMENTAL A ESCALA.....	37
4.1. Hardware del Prototipo a escala.....	37
4.2. Software: Programación del Hardware.....	43
4.3. Programación del Prototipo en MikroBasic.....	45
CAPÍTULO 5: CONCLUSIONES Y RECOMENDACIONES.....	53
5.1. Conclusiones.....	53
5.2. Recomendaciones.....	53
REFERENCIAS BIBLIOGRÁFICAS.....	55

Índice de Figuras

Capítulo 2

Figura 2. 1: Microcontrolador PIC.....	5
Figura 2. 2: Familia de los Microcontrolador PIC.....	6
Figura 2. 3: Mapa de memoria de programa.	8
Figura 2. 4: Mapa de la memoria de datos SRAM.....	12
Figura 2. 5: Subsistema de entrada del temporizador.....	15
Figura 2. 6: Subsistema de salida del temporizador.....	16
Figura 2. 7: Configuración Maestro/Esclavo.....	18
Figura 2. 8: Conexión a un oscilador externo.	20
Figura 2. 9: Características básicas de Basic.....	21
Figura 2. 10: Representación de identificadores en MikroBasic.....	22
Figura 2. 11: Representación de etiquetas Basic.	22
Figura 2. 12: Representación en MikroBasic de una etiqueta.	23
Figura 2. 13: Sintaxis y ejemplos en MikroBasic de una constante.....	24
Figura 2. 14: Sintaxis y ejemplos en MikroBasic de una variable.....	24
Figura 2. 15: Sintaxis y ejemplos en MikroBasic de símbolos.....	25

Capítulo 3

Figura 3. 1: Modo de propagación superficial.....	29
Figura 3. 2: Modo de propagación aérea.	30
Figura 3. 3: Modo de propagación mediante línea de vista.....	30

Capítulo 4

Figura 4. 1: Tarjeta de control con PIC16F886.....	37
Figura 4. 2: Tarjeta del Puente H.	38
Figura 4. 3: Tarjeta del sensor de distancia ultrasónico.	38
Figura 4. 4: Diagrama de bloques de interconexión entre dispositivos electrónicos del prototipo.	42
Figura 4. 5: Programador P.PICI&T04.	42
Figura 4. 6: Pasos para la programación.	43

Figura 4. 7: Descripción de pines del PIC16F886.	44
Figura 4. 8: Conexión entre el programador y módulo de entrenamiento.	44
Figura 4. 9: Diagrama ASM para configuración UART.....	45
Figura 4. 10: Diagrama ASM para subrutina avanzar adelante.....	46
Figura 4. 11: Diagrama ASM para subrutina avanzar atrás.	46
Figura 4. 12: Interfaz del computador AccessPort.....	47
Figura 4. 13: Vista Superior del Robot a Escala.....	47
Figura 4. 14: Vista Interior de tarjetas electrónicas del robot a escala.	48

Índice de Tablas

Capítulo 2.

Tabla 2. 1: Microcontroladores PIC que usan Arquitectura Harvard.	7
Tabla 2. 1: Códigos binarios para acceder al registro de estados.....	13

Capítulo 3.

Tabla 3. 1: Clasificación de las Bandas de Frecuencia del Espectro Radioeléctrico.....	27
Tabla 3. 2: Lista de bandas disponibles en la Clasificación de las Bandas de Frecuencia del Espectro Radioeléctrico	28
Tabla 3. 3: Rango de frecuencias para enlaces de redes inalámbricas	28
Tabla 3. 4: Características del estándar Home RF.....	34

Capítulo 4.

Tabla 4. 1: Interconexión entre dispositivos.	39
Tabla 4. 2: Características de la tarjeta de control.	39
Tabla 4. 3: Configuración del puente H.	40
Tabla 4. 4: Configuración pines entre tarjeta de control y antena XBEE.	40
Tabla 4. 5: Configuración entre tarjeta de control y sensor ultrasónico.....	41

Resumen

TITULO

“DISEÑO E IMPLEMENTACIÓN A ESCALA DE UN ROBOT ACUÁTICO RADIOCONTROLADO MEDIANTE COMUNICACIÓN INALÁMBRICA”

DESCRIPCION

En el cual se conoció la problemática a investigar, dando como resultado la aplicación de un robot acuático radiocontrolado y operado mediante una computadora de escritorio o portátil.

Lo novedoso fue que hasta la presente no se ha visto ni encontrado trabajos de grado o titulación similares en la Carrera de Ingeniería en Telecomunicaciones.

En el Capítulo 1, se detalla las Generalidades del Trabajo de Titulación entre los más destacados son: definición del problema, objetivos tanto el general como los específicos, y la metodología empleada.

En el Capítulo 2, se describe el Estado del Arte de Sistema Microcontroladores PIC en Robótica, lo cual ayudó para el desarrollo del robot a escala acuático.

En el Capítulo 3, se describe el medio de transmisión utilizado para ser controlado inalámbricamente desde un computador de escritorio o portátil.

En el capítulo 4, se realiza el Desarrollo Experimental a Escala del Robot Acuático en la cual se muestra los pasos para programar en Basic y en AppInventor.

En el Capítulo 5, se presenta las conclusiones y recomendaciones.

ABSTRACT

TITLE

“DESIGN AND IMPLEMENTATION OF AN SCALE RADIO-CONTROLLED ACUATIC ROBOT THROUGH WIRELESS COMMUNICATION”

DESCRIPTION

Where the problem to investigate is known, giving as result the application of a radio-controlled water robot operated through a desktop or laptop.

What is new is that so far there has not been seen or found work degrees or similar theme in the telecommunication engineering career.

Chapter 1, details general work degree between the most important: problem definition, general and specific objectives, and the applied methodology.

Chapter 2, describes the state of the art PIC Microcontrollers system in Robotics, which helped to the development of the aquatic scale robot.

Chapter 3, describes the transmission channels used to be controlled through a desktop or laptop wirelessly.

Chapter 4, experimental development of the aquatic scale robot in which the steps for basic programming and in Appinventor are shown.

Chapter 5, conclusion and recommendations.

CAPÍTULO 1: GENERALIDADES DEL TRABAJO DE TITULACIÓN

1.1. Introducción.

La navegación no es una tecnología pero es un arte antiguo para cursar mares, siendo una de las actividades importantes para el desarrollo de importaciones y exportaciones, que beneficien a una sociedad. Durante muchos años por no decir siglos de constantes esfuerzos han permitido pasar de simples balsas flotantes a las complejas embarcaciones actuales.

Obviamente al ser un arte antiguo el navegar, estos en la actualidad son automatizados y controlados por sistemas electrónicos avanzados. Con esto no quiere decir que el control está a cargo de toda la embarcación, sino que sigue en manos del ser humano, simplemente que estos sistemas electrónicos inteligentes son para ayuda en maniobras marítimas.

Aunque en ciertas embarcaciones marítimas existen equipos de control con realimentación, para lo cual se enumeran ejemplos de sistemas de control disponibles comercialmente como: pilotos automáticos para barcos y vehículos submarinos para el control de giro y mantenimiento del rumbo. O incluso sistemas de control de torpedos.

1.2. Antecedentes.

El despliegue de aplicaciones de robótica a través de los microcontroladores ha evolucionado en la última década, tal es así que en países europeos y del continente americano las Instituciones de Educación Superior centran sus actividades de investigación en proyectos finales de carrera en el desarrollo de aplicaciones a escala de robots autónomos y no autónomos (controlados), donde la idea principal es que la microelectrónica es aplicable en cualquier rama de las ciencias exactas y ciencias aplicadas a la ingeniería.

1.3. Justificación del Problema.

El despliegue de dispositivos electrónicos aplicados a diferentes áreas, en particular para el presente trabajo de titulación es la navegación mediante un Robot Acuático Radiocontrolado a escala. La tarea no es tan compleja porque la embarcación a escala se la puede conseguir en cualquier almacén de juguetes, lo más importante es el diseño electrónico mediante los microcontroladores PIC y su respectiva programación en lenguaje de alto nivel para ser radiocontrolado.

Con este dispositivo electrónico se controlará la embarcación mediante maniobras u operaciones marítimas, ya que esto es una manera de mirarlos desde el lado de la robótica, donde la cooperación entre agentes dinámicos está atrayendo el interés investigador desde hace años.

1.4. Definición del Problema.

Necesidad de diseñar e implementar un robot acuático Radiocontrolado que permita la manipulación de embarcaciones como barcos remolcadores que agilicen el traslado de embarcaciones de Grandes tamaños, en general para ayudar a movilizarse en los mares sin problemas a encallarse o tener rupturas en su casco

1.5. Objetivos del Problema de Investigación.

1.5.1. Objetivo General.

Implementar el diseño de un Robot Acuático Radiocontrolado para el Laboratorio de Electrónica de la Carrera de Ingeniería en Telecomunicaciones de la Facultad de Educación Técnica para el Desarrollo (FETD).

1.5.2. Objetivos Específicos.

- Describir el Estado del Arte de microcontroladores a través de aplicaciones en robots autónomos y no autónomos.

- Elaborar la programación en MikroBasic para controlar el hardware y software del barco acuático.
- Implementar el Robot Acuático Radiocontrolado para futuros concursos de robótica que se realizan cada año a nivel nacional a través del CER.

1.6. Idea a Defender.

Mediante la implementación del Robot Acuático Radiocontrolado permitirá desarrollar y mejorar el nivel investigativo de los estudiantes de la carrera de Ingeniería en Telecomunicaciones, es decir, que los más favorecidos serán los estudiantes para presentar nuevas propuestas e innovadoras en temas de titulación, proyectos de investigación del SINDE y de maestría.

1.7. Metodología de Investigación.

De acuerdo al tema propuesto de titulación, se procedió a determinar que el paradigma de investigación es del tipo Empírico – Analítico, con enfoque cuantitativo, es decir, que se definiría como de carácter:

- a. Exploratorio, debido a que explora los microcontroladores cuya aplicabilidad es en diferentes campos de estudio, que permite determinar las herramientas para proceder a la implementación del presente trabajo de titulación.
- b. Descriptivo, ya que a través de la programación en MikroBasic se puede ejecutar lo expuesto en el carácter exploratorio.
- c. Explicativo, porque al final del proyecto es ejecutable para su correcta explicación.

CAPÍTULO 2: Estado del Arte de Sistema Microcontroladores PIC en Robótica.

Para el presente capítulo se presentará una descripción general de conceptos básicos que hacen referencia al trabajo de titulación mediante un estudio exploratorio del estado del arte de los sistemas microcontroladores PIC aplicados a la robótica.

2.1. ¿Qué es un Microcontrolador?

Un microcontrolador es esencialmente un equipo de un solo chip de bajo costo. Chip único significa que todo el sistema informático se encuentra dentro de los límites de una pastilla de silicona encapsulada dentro de la carcasa de plástico de un circuito integrado. El microcontrolador tiene características similares a las de un ordenador personal estándar.

El microcontrolador contiene una CPU (Unidad Central de Procesamiento), la RAM (Memoria de Acceso Aleatorio), ROM (Memoria de Sólo Lectura), pines de I/O (Entrada/Salida), Puertos Seriales y Paralelos, temporizadores, y en ocasiones otros periféricos integrados tales como convertidores A/D (Analógico/Digital) y convertidores D/A (Digital/Analógico), tal como se ilustra en la figura 2.1. La característica clave, sin embargo, es la capacidad del microcontrolador de subir, almacenar y ejecutar un programa.

Algo similar lo define Reyes, C. A. (2008) indicando que un microcontrolador es un circuito integrado, en la que internamente dispone de una arquitectura similar a un computador, esto es CPU, memorias RAM, EEPROM, y circuitos de entrada y salida.

Según Bates, M. (2013) indica que aunque pequeño, los microcontroladores son complejos, y tenemos que prestar atención a la forma

en que el hardware y el software (programa de control) trabajan juntos para comprender los procesos en el trabajo.

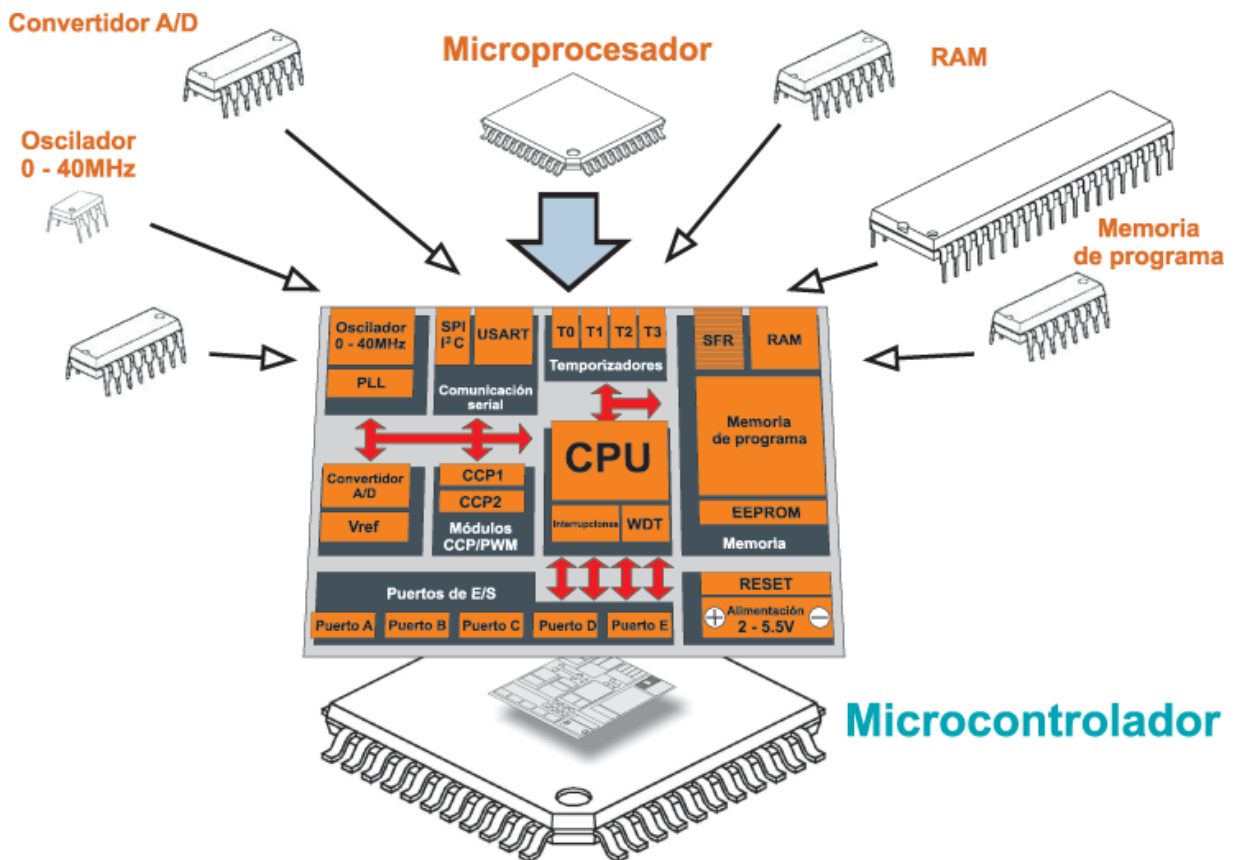


Figura 2. 1: Microcontrolador PIC.
Fuente: Verle, M. (2010)

2.2. ¿Por qué usar un Microcontrolador?

Siendo ordenadores mono pastilla baratos, microcontroladores son fáciles de integrar en mayor diseños de circuitos electrónicos. Su capacidad para almacenar y ejecutar programas únicos hace extremadamente versátil. Por ejemplo, se puede programar un microcontrolador para tomar decisiones y realizar funciones sobre la base de situaciones (la lógica de la línea I / O) y eventos.

Las funciones matemáticas y lógicas permiten al microcontrolador para imitar la lógica sofisticada y circuitos electrónicos.

2.3. El Microcontrolador PIC

La familia PIC tiene muchos primos con varias características únicas, tal y como se ilustra en la figura 2.2. A pesar de esto, Microchip diseñó los PIC's de compartir muchos atributos comunes tales como diseño de la memoria y el diseño de envases. Por ejemplo, los pines de salida para el PIC de paquete de 18 pines, es el mismo si se trata de un dispositivo con E/S estándar o comparadores analógicos, e inclusive los puertos para convertidores tanto analógico/digital como digital/analógico.

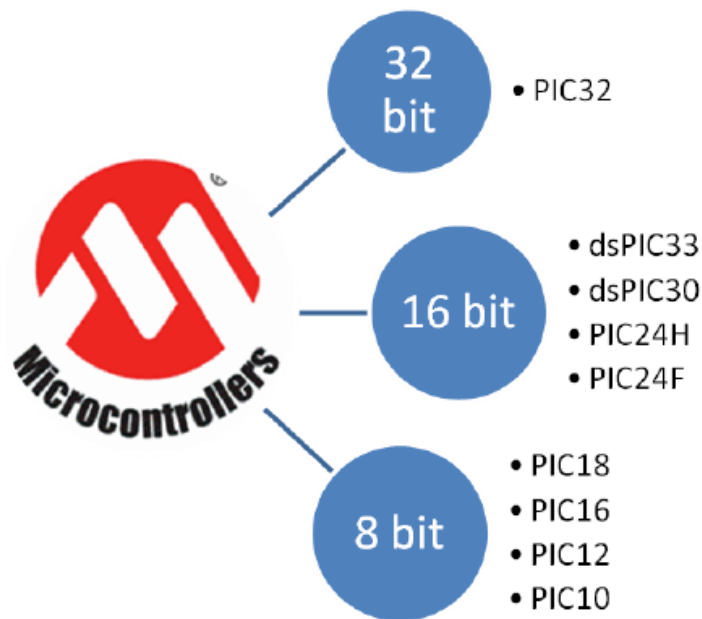


Figura 2. 2: Familia de los Microcontrolador PIC.
Fuente: Autor.

Todos los microcontroladores PIC implementan la arquitectura Harvard, lo que significa que su memoria de programa está conectada a la CPU a través de más de 8 líneas. Dependiendo del ancho de bus, hay microcontroladores de 12, 14 y 16.

En la tabla 2.1 se muestra las principales características de estas tres categorías de los microcontroladores PIC.

Tabla 2. 1: Microcontroladores PIC que usan Arquitectura Harvard.

Family	ROM [Kbytes]	RAM [bytes]	Pins	Clock Freq. [MHz]	A/D Inputs	Resolution of A/D Converter	Comparators	8/16-bit Timers	Serial Comm.	PWM Outputs	Others
Base-Line 8-bit architecture, 12-bit Instruction Word Length											
PIC10FXXX	0.375 - 0.75	16 - 24	6 - 8	4 - 8	0 - 2	8	0 - 1	1 × 8	-	-	-
PIC12FXXX	0.75 - 1.5	25 - 38	8	4 - 8	0 - 3	8	0 - 1	1 × 8	-	-	EEPROM
PIC16FXXX	0.75 - 3	25 - 134	14 - 44	20	0 - 3	8	0 - 2	1 × 8	-	-	EEPROM
PIC16HVXXX	1.5	25	18 - 20	20	-	-	-	1 × 8	-	-	Vdd = 15V
Mid-Range 8-bit architecture, 14-bit Instruction World Length											
PIC12FXXX	1.75 - 3.5	64 - 128	8	20	0 - 4	10	1	1 - 2 × 8 1 × 16	-	0 - 1	EEPROM
PIC12HVXXX	1.75	64	8	20	0 - 4	10	1	1 - 2 × 8 1 × 16	-	0 - 1	-
PIC16FXXX	1.75 - 14	64 - 368	14 - 64	20	0 - 13	8 or 10	0 - 2	1 - 2 × 8 1 × 16	USART I2C SPI	0 - 3	-
PIC16HVXXX	1.75 - 3.5	64 - 128	14 - 20	20	0 - 12	10	2	2 × 8 1 × 16	USART I2C SPI	-	-
High-End 8-bit architecture, 16-bit Instruction Word Length											
PIC18FXXX	4 - 128	256 - 3936	18 - 80	32 - 48	4 - 16	10 or 12	0 - 3	0 - 2 × 8 2 - 3 × 16	USB2.0 CAN2.0 USART I2C SPI	0 - 5	-
PIC18FXXJXX	8 - 128	1024 - 3936	28 - 100	40 - 48	10 - 16	10	2	0 - 2 × 8 2 - 3 × 16	USB2.0 USART Ethernet I2C SPI	2 - 5	-
PIC18FXXKXX	8 - 64	768 - 3936	28 - 44	64	10 - 13	10	2	1 × 8 3 × 16	USART I2C SPI	2	-

Fuente: Verle, M. (2010)

2.3.1. Memoria de Programa.

Según Ángulo U., J. M. & Ángulo M., I. (2003) indica que los microcontroladores son construidos para que la memoria de programa se almacene cada una de las instrucciones del programa de control, con la ventaja de no utilizar memorias externas para ampliaciones como ocurre con los microprocesadores.

La memoria de programa es el espacio donde reside el programa desarrollado en lenguaje de alto nivel, para este caso el PicBasic (PBP). Al leer una hoja de datos y ver el PIC es 0.5K o 1k o 2k, se refieren al espacio de memoria del programa. Los tamaños <0.5K, 1k, 2k, 4k, y 8k> son los tamaños de memoria de programa para la mayoría de los dispositivos PIC. Esto puede parecer muy pequeño, pero para las funciones de un PIC diseñados para hacer esto.

En la figura 2.3 se muestra el diagrama de la memoria de programa de un microcontrolador PIC.

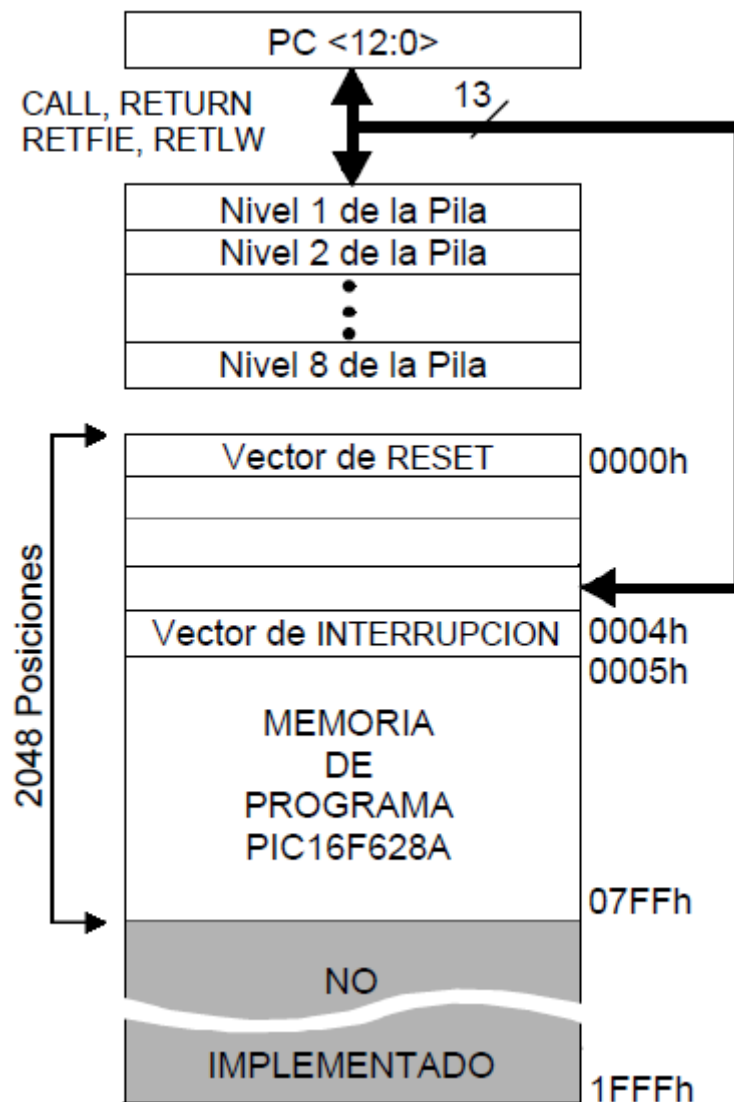


Figura 2. 3: Mapa de memoria de programa.
Fuente: Reyes, C. A. (2008)

Un programa PBP puede tener varios cientos de comandos y todavía caben en un dispositivo 1k. Debido a que cada comando de PBP es tan poderoso, un programa 1k podía controlar un motor con realimentación y control de la dirección, o el monitor de un sistema de alarma antirrobo y se mostrará información en una pantalla de cristal líquido (LCD). Adicionalmente, un puerto serial envía el estado de los sensores del sistema a un PC, a través de un solo comando conocido como SEROUT.

A continuación se muestran los tipos de memoria de un microcontrolador PIC:

a. Memoria ROM (memoria de solo lectura) enmascarada.

Según Maini, A. K. (2007) la memoria ROM contiene todas las instrucciones de programa y los datos constantes.

b. Memoria EPROM.

Según Ángulo U., J. M. et al. (2003) la grabación de una memoria EPROM es a través de un software instalado en una PC, siendo esta muy interesante en el diseño y compilación de programas, aunque es elevado su costo.

c. Memoria OTP.

La memoria OTP es aquella que puede ser grabada por única vez por el programador o usuario, muy similar a la memoria EPROM aunque después no se pueda borrar lo grabado.

d. Memoria EEPROM.

Según lo indicado por Sánchez, J. & Cantón, M. P. (2007), la EEPROM (doble e-PROM) cuyo significado indica que es una memoria de sólo lectura, programable y borrable eléctricamente. La memoria EEPROM se emplean o disponen en los computadores de escritorio o portátiles, y dispositivos digitales; todos estos elementos son de almacenamiento no volátil.

En si la memoria EEPROM no es una memoria RAM, como sabemos las RAM son volátiles mientras que la EEPROM suspende sus datos al mismo tiempo que se desconecta la alimentación. En otras palabras, las EEPROM funcionan actualmente en todas las unidades flash USB y como almacenamiento no volátil de miles de microcontroladores de cualquier fabricante.

e. Memoria FLASH.

La memoria Flash es conocida como la sucesora de la EPROM UV, se hizo en los años 80 en los laboratorios de INTEL. Desde esa memoria puede ser reescrito prácticamente un número ilimitado de veces, los microcontroladores con memoria Flash ROM son ideales para el aprendizaje, la experimentación y la producción a pequeña escala. Debido a su gran popularidad, la mayoría de los microcontroladores son hoy fabricados en tecnología flash. Por lo tanto, si usted va a comprar un microcontrolador, el tipo que debe de buscar es definitivamente memoria flash.

2.3.2. Memoria de Datos.

La estructura y organización de la memoria de datos en el hardware PIC también tiene algunas características únicas e interesantes. El programador acostumbrado al espacio de memoria plana, direccionable de la computadora von Neumann con sus múltiples registros de la máquina puede requerir algún tiempo para familiarizarse con los formatos de datos de los PIC's.

Según Hellebuyck, C. (2003) la memoria de datos es donde se almacenan todas las variables, siendo la memoria RAM (Random Access Memory), lo que significa que el PIC se desconecta la energía inmediatamente se pierde toda la memoria de datos. La memoria de datos es de 8 bits de ancho, por lo que los países insulares del Pacífico son considerados los microcontroladores de 8 bits.

La memoria de datos PIC también viene al igual que la memoria del programa. La primera sección de la memoria de datos está reservada para los registros de funciones especiales. Estos registros son los lugares clave que controlan la mayor parte de las funciones dentro de la PIC.

Mientras que para Ángulo U., J. M. et al. (2003) los datos varían continuamente por los datos contenidos deberían para lectura y escritura (W/R), es decir, que la memoria más adecuada es la SRAM (memoria estática) aunque sea volátil.

A continuación se describen los registros w y datos:

a. Registro w:

Los PIC's tienen sólo un registro direccionable llamado registro w, el programador CISC está acostumbrado a tener múltiples registros de propósito general, en la que los datos se pueden mover y posteriormente recuperarlos, tienen que acostumbrarse a un registro único que participa en prácticamente todas las instrucciones de un equipo. Añádase a esto la falta de una pila direccionable en la que los datos pueden ser empujados y saltados, y así la programación de los PIC's serían un paradigma diferente.

b. Registro de datos:

La memoria de datos de los microcontroladores PIC's se compone de registros, también llamados registros de datos. Estos se comportan más como las variables convencionales, y pueden ser dirigidas directamente e indirectamente. Todos los registros de datos son de 8 bits. Los registros de datos pueden ser de dos tipos: registros de propósito general (GPRS) y registros de funciones especiales (SFR).

Según Reyes, C. A. (2008) la RAM dispone de 512 líneas cada una de 8 bits y particionada en 4 bancos que son: banco 0, banco 1, banco 2 y banco 3

con 128 bytes cada uno; estos se acceden a través de los bits RP1 y RP0 del registro STATUS tal como se muestra en la figura 2.4.

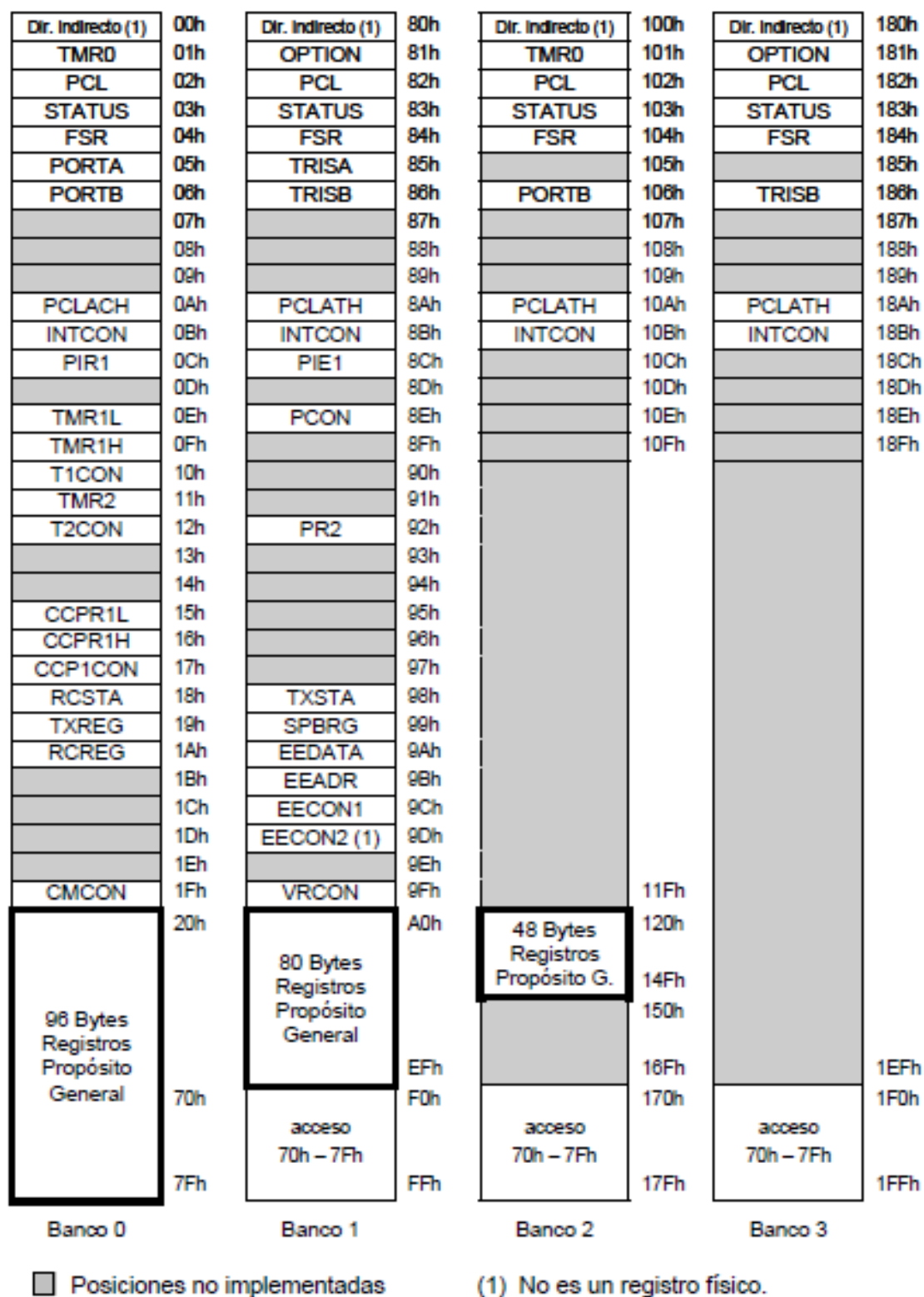


Figura 2. 4: Mapa de la memoria de datos SRAM.
Fuente: Reyes, C. A. (2008).

En cambio para Ángulo U., J. M., Romero Y., S. & Ángulo M., I. (2006) indica la manera de acceder a la memoria RAM mediante los bits 6 y 5 (registros de estados) también conocidos como RP1 y RP0, estos códigos se muestran en la tabla 2.2.

Tabla 2. 2: Códigos binarios para acceder al registro de estados.

BANCO	RP1	RP2
0	0	0
1	0	1
2	1	0
3	1	1

Elaborado: Autor

2.3.3. Líneas de E/S para módulos periféricos.

En los PIC's de gama media se tienen módulos especiales para implementar periféricos y funciones de E/S. Mientras más complejo es el dispositivo (PIC), los módulos periféricos son más probables de su presencia. Por ejemplo, un simple PIC de gama media como el 16F84A contiene unos módulos periféricos, específicamente, la memoria de datos EEPROM, puertos E/S, y un módulo del temporizador.

Mientras que para los PIC's 16F87x, por otro lado, además de los puertos E/S, memoria EEPROM, y tres temporizadores individuales, tienen un puerto esclavo en paralelo, un módulo WPM (captura y comparador), un módulo MSSP (master puerto serie síncrono), un USART (asincrónico universal/receptor y el transmisor síncrono) del módulo, y un módulo convertidor A/D (convertidor analógico-digital).

A continuación se describen ciertos componentes periféricos:

a. Convertidores Analógico/Digital y Digital/Analógico.

Los convertidores A/D (Analógico/Digital) y D/A (Digital/Analógico), poseen una interfaz con dispositivos analógicos. Por ejemplo, el convertidor analógico/digital proporciona una interfaz entre el microcontrolador y los sensores que producen equivalentes eléctricas analógicas de los parámetros físicos reales que deben ser controlados.

El convertidor de digital a analógica, por otra parte, proporciona una interfaz entre el microcontrolador y los accionadores que proporcionan la función de control. A modo de ejemplo, tanto 68HC11 y 68HC12 de Freescale Semiconductor tienen ocho canales, ocho bits de analógico/digital. La función de conversión digital/analógico en los microcontroladores, es proporcionada por una combinación de modulador de ancho de pulso (PWM) seguido por un filtro. Como un ejemplo, el chip 68HC12 tiene un PWM 16-bits/dos canales.

b. Puertos de E/S.

Los puertos de E/S proporcionan una interfaz entre el microcontrolador y los periféricos de E/S, tales como: teclados, pantallas (displays, LCD), etc. La familia de microcontroladores PIC16F84A tiene cuatro puertos de ocho bits de E/S, mientras que para el microcontrolador PIC16F877A ofrece 38 de propósito general E/S pines incluyendo 16 pines bidireccionales de E/S, 11 pines sólo de entrada y 11 pines sólo salida.

c. Contadores/Temporizadores (*Counters/Timers*).

Los contadores/temporizadores suelen realizar tres funciones, que son: **(1)** controlar el tiempo y/o medir el intervalo de tiempo entre eventos, **(2)** cuenta el número de eventos y **(3)** generar velocidades de transmisión de los puertos serie. El microcontrolador popular

PIC16F84, tiene un temporizador de 8 bits y un contador prescalador de 8 bits.

En la figura 2.5 se muestra la disposición esquemática mediante un diagrama de bloques, donde el contador captura el evento de tiempo de entrada en la forma de su contenido en el momento de la ocurrencia del evento. De hecho, el contador captura el tiempo relativo del evento ya que el contador es de funcionamiento libre. Valores de tiempo absolutos se pueden calcular a partir de los valores del reloj del sistema relativos. Como ejemplo, considere un microcontrolador con un reloj de 10 MHz y un subsistema de contadores/temporizadores de 16 bits. Este contador se tardará 6,5536 ms en contar de 0000 a FFFF (notación hexadecimal). Supongamos que se desea encontrar la frecuencia de una señal periódica cuyos flancos sucesivos ascendentes o descendentes se observan a ocurrir en 0010 y 0150 respectivamente, lo que corresponden a 16 y 336 en decimal. Por lo tanto, el intervalo de tiempo entre dos flancos sucesivos son iguales a $320 \times 0.1 = 32 \mu s$, siendo finalmente la frecuencia $\left(\frac{1}{32}\right) MHz = 31.25 kHz$.

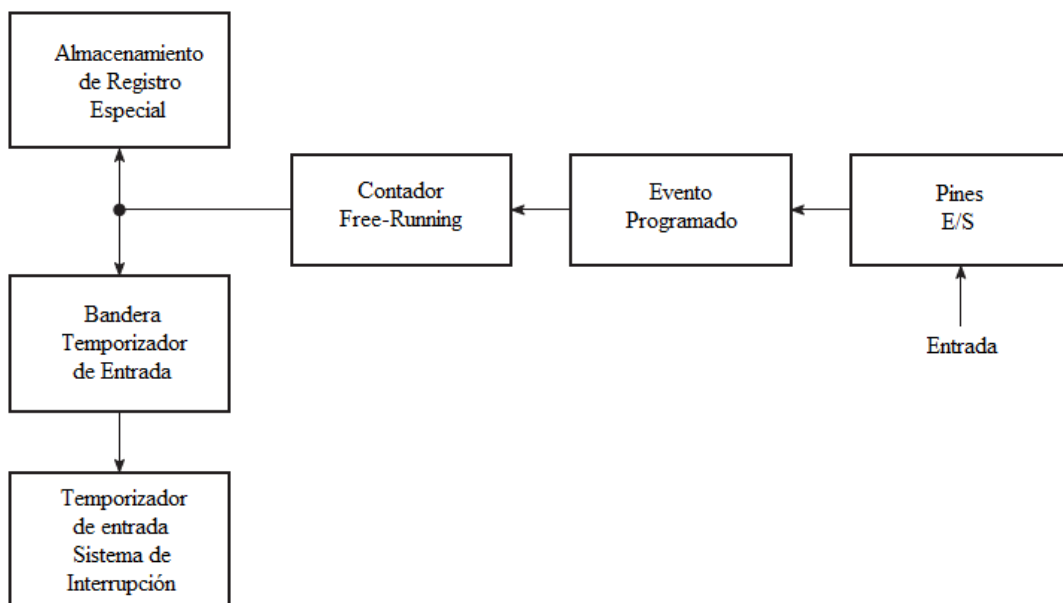


Figura 2. 5: Subsistema de entrada del temporizador.
Fuente: Reyes, C. A. (2008).

En la figura 2.6 se muestra una disposición esquemática del diagrama de bloques de salida del temporizador, el mismo explica una vez más los valores del contador en marcha libre, permitiendo sincronizar la hora de los cambios de estados deseados de la lógica en el pin de salida. Esta función también se puede utilizar para generar un pulso aperiódico o una señal periódica de cualquier ciclo de trabajo deseado.

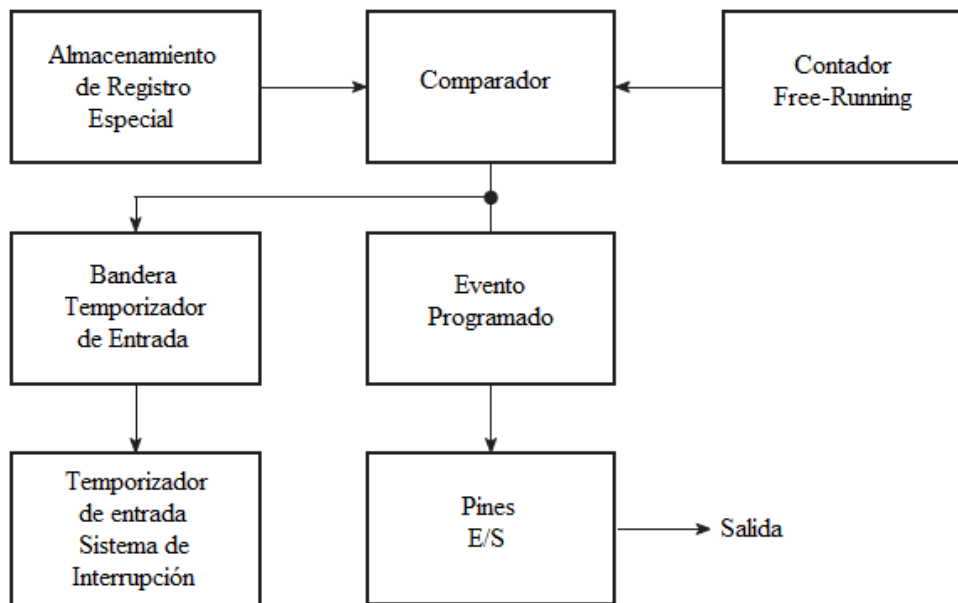


Figura 2. 6: Subsistema de salida del temporizador.
Fuente: Reyes, C. A. (2008).

d. Interface de Comunicación Serial.

Existen dos tipos de interfaz de comunicación serial, a saber, la interfaz de comunicación asíncrona y la interfaz de comunicación síncrona. La interfaz de comunicación asíncrona utiliza un protocolo de inicio y un bit de parada, para sincronizar el transmisor y el receptor. Los bits de inicio y parada están incrustados en cada byte de datos.

En comparación con la interfaz de comunicación síncrona, que ofrece velocidades de transmisión de datos más bajas. También se conoce como el receptor/transmisor universal asíncrono (UART) o la interfaz

de comunicación en serie (SCI). La interfaz de comunicación síncrona utiliza un reloj sincronizado para transmitir y recibir cada bit. La sincronización de los relojes del transmisor y receptor se logra generalmente mediante el uso de una línea de reloj adicional que une el transmisor y el receptor. No se recomienda para las comunicaciones de larga distancia.

e. Bus de Circuito Interintegrado (I²C)

El circuito Interintegrado (I²C) es un bus de dos hilos, baja a la interfaz de comunicación en serie a velocidad media, desarrollada por Philips Semiconductors a principios de 1980 para comunicaciones de chip a chip.

Los dos cables del bus I²C se llaman reloj (SCL) y de datos (SDA). El alambre de SDA lleva datos, mientras que el cable SCL sincroniza el transmisor y el receptor durante la transferencia de datos. Se trata de un protocolo de comunicación estándar de la industria, probada en una variedad de productos electrónicos, que es particularmente facilitada por su bajo coste y características de gran alcance.

Es apoyado por un gran número de fabricantes de semiconductores y sistemas que ofrecen una variedad de productos electrónicos, incluyendo dispositivos de entrada y salida, de diferentes tipos de sensores, dispositivos de memoria, pantallas, dispositivos de entrada de datos, etc. Algunas de las características importantes que ofrecen dispositivos I²C se describen brevemente en los siguientes párrafos.

Los dispositivos I²C ofrecen jerarquía maestro/esclavo. Estos se clasifican como maestro (el dispositivo que inicia el mensaje) o esclavo (el dispositivo que responde al mensaje). El dispositivo puede ser único maestro o sólo esclavo, se puede cambiar entre maestro y esclavo dependiendo de los requisitos de la aplicación.

Una configuración maestro/esclavo es posible, en la que un maestro (por ejemplo, un microcontrolador) está conectado a muchos chips configurados como esclavos, tal como se muestra en la figura. 2.7. Cada uno de los dispositivos esclavos I2C es identificable por una única dirección. Cuando el dispositivo maestro envía un mensaje, que incluye la dirección del dispositivo esclavo deseado al principio del mensaje.

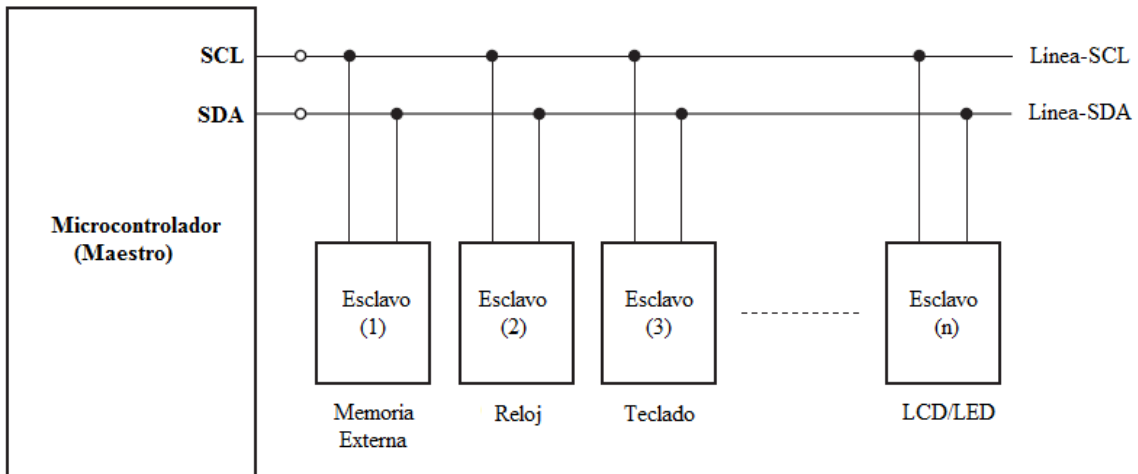


Figura 2. 7: Configuración Maestro/Esclavo.
Fuente: Reyes, C. A. (2008).

f. Buses CAN (*Controller Area Network*).

El Bus CAN conocido como controladores redes, es una robusta interfaz de comunicación serial, se utiliza en una amplia gama de sistemas empotrados o embebidos, así como las aplicaciones de control de automatización. Fue introducido por Bosch en 1986 para redes a bordo de vehículos.

El protocolo CAN fue estandarizado internacionalmente en 1993 como la norma ISO-11898-1 y comprende la capa de enlace de datos del modelo de referencia ISO/OSI de siete capas. El protocolo proporciona dos servicios de comunicación, a saber, la transmisión de tramas de datos (envío de un mensaje) y de solicitud de transmisión remota (solicitante de un mensaje).

Todos los otros servicios, como la señalización de error, la retransmisión automática de tramas erróneas, etc., son realizados por los chips CAN. Algunas de las características importantes del protocolo CAN son los siguientes: **(1)** Proporciona una jerarquía de varios maestros, esto permite al usuario construir sistemas inteligentes y redundantes. **(2)** Se utiliza el método de comunicación de difusión, donde el remitente de un mensaje transmite a todos los dispositivos conectados al bus.

g. Buses LIN (*Local Interconnect Network*).

El Bus LIN conocida como red de interconexión local, es una red de bus serial de difusión que se utiliza como una subred de bajo coste que un bus CAN para integrar sensores o actuadores inteligentes en los automóviles modernos.

Se compone de un maestro (típicamente un microcontrolador moderadamente potente) y hasta 16 esclavos (microcontroladores menos potentes, o más baratos como el ASIC). No ofrece una función de detección de colisiones y, por tanto, todos los mensajes son iniciados por el maestro, con a lo sumo un esclavo en respuesta a un identificador de mensaje dado.

Varias de estas redes LIN pueden estar vinculadas a una red de capa superior de la CAN a través de sus respectivos dueños.

2.4. Programación MikroBasic para PIC's.

Antes de describir como es la programación en MikroBasic, debo de indicar que existen dos tipos de lenguaje de programación (véase la figura 2.8), el primero de bajo nivel (lenguaje de máquina) que se programa en Assembler (ensamblador) y finalmente de alto nivel que se puede programar en Basic, Pascal o C, lo de MikroBasic es simplemente un compilador profesional, lo

mismo que PicBasic, sino que toma el nombre por la empresa Mikroe (MikroElektronika) dueña del compilador MikroBasic.

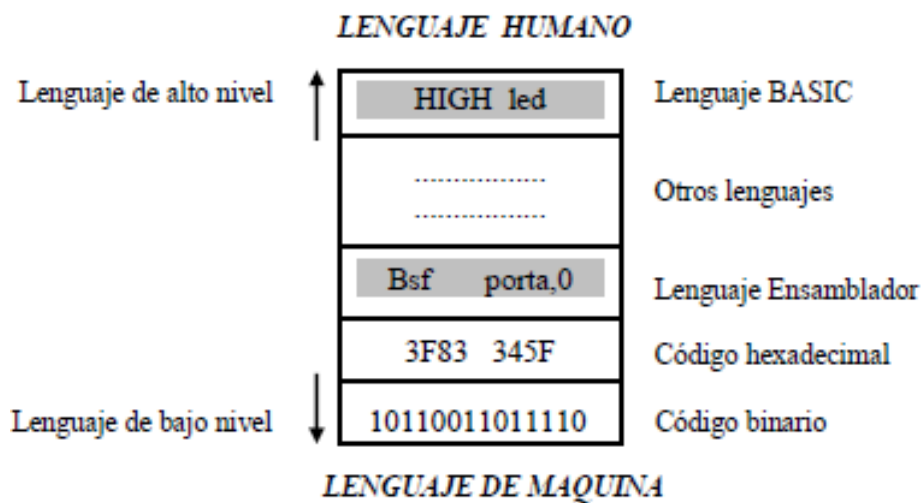


Figura 2. 8: Conexión a un oscilador externo.
Fuente: Reyes, C. A. (2008).

Para la descripción de la programación en MikroBasic se utiliza información de textos, pero en su mayoría de Verle, M. (2010).

2.5. Características de programación en Basic.

Programar en Basic es muy simple, sencillo y de fácil aprendizaje, pero para esto los usuarios o programadores deben conocer de algunos elementos que son básicos del programa Basic, tales como: comentarios, directivas, entre otras, etc.

De acuerdo a Verle, M. (2010), estas características son esenciales a la hora de programar en Basic, independientemente del compilador a utilizar, ya sea PicBasic (PBC) o MikroBasic. En la figura 2.9 se muestran algunas de las características de la programación Basic, ya indicada anteriormente.

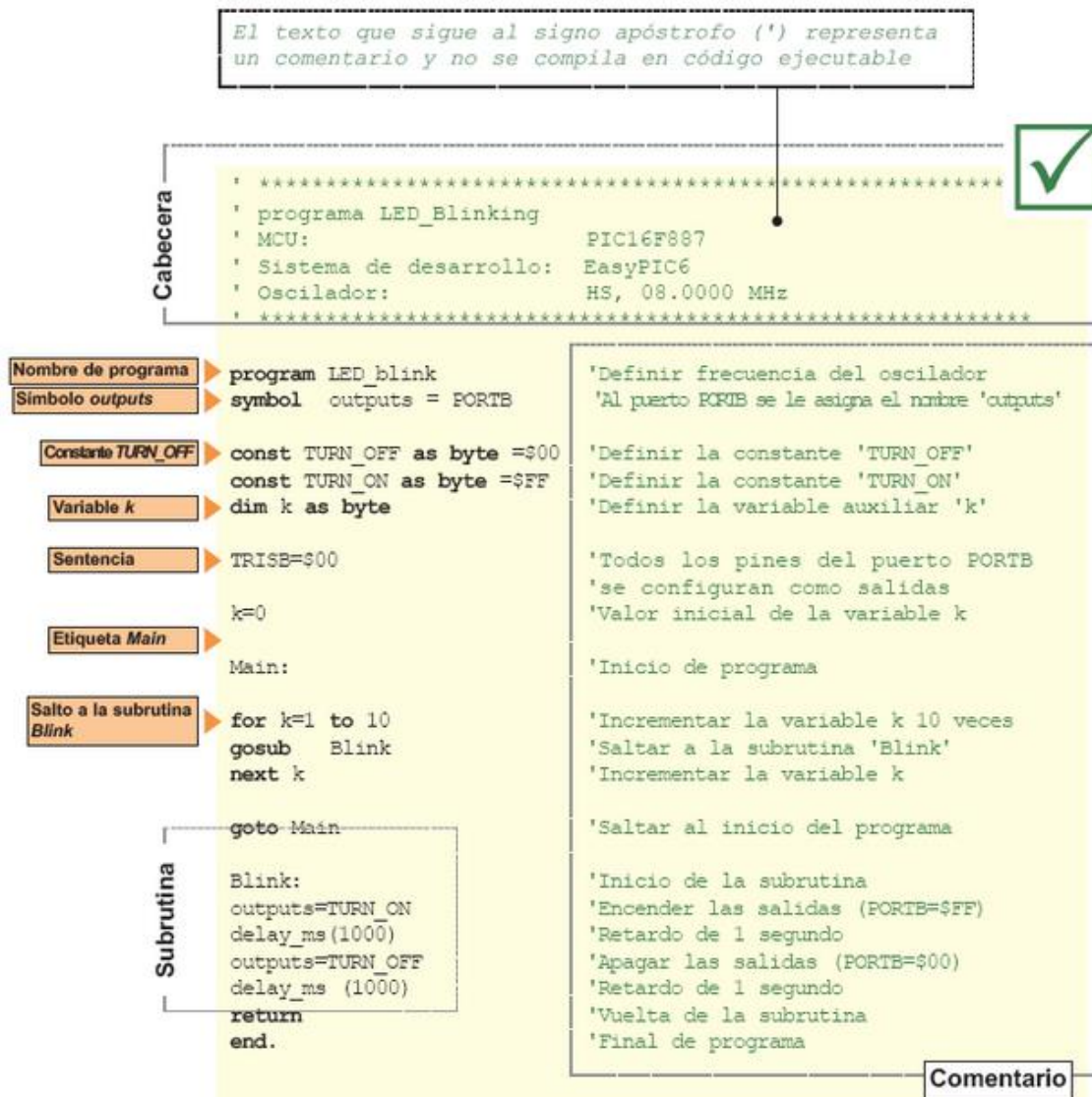


Figura 2. 9: Características básicas de Basic.
Fuente: Verle, M. (2010)

2.5.1. Identificadores.

El identificador representa el nombre de algún elemento MikroBasic. Los identificadores se utilizan en MikroBasic para firmar líneas de programa y los nombres de varios símbolos. Un identificador podría ser cualquier cadena de letras, números o guiones incluso con el límite de que no se le permite comenzar con un número. Los identificadores no distinguen mayúsculas y minúsculas, por lo que el RESET y Reset (véase figura 2.10) son tratados de la misma manera. La duración máxima de este tipo de cadenas es de 32 caracteres.

```
symbol Reset = PORTA.0      ' RA0 se identifica como Reseteo
symbol Led_0 = PORTB.0      ' RB0 se identifica como LED0
```

Figura 2. 10: Representación de identificadores en MikroBasic.
Fuente: Autor

2.5.2. Etiquetas.

Las etiquetas representan el signo textual para alguna línea de programación o algunos de sus fragmentos respectivamente en los que el programa puede saltar a través de algunas de las instrucciones (véase la figura 2.11) que se utilizan para cambiar un flujograma (diagrama ASM), siendo obligatorio para terminar con la etiqueta. Al contrario de muchas versiones viejas de Basic, MikroBasic no permite valores numéricos como etiquetas.

```
symbol Taster = PORTA.0
symbol LED_0 = PORTB.0

B0 var byte

Main:           ' Label Main
    B0 = 0
    button Set,0,255,0,B0,1,LED_toggle
    goto Main

LED_toggle:    ' Label LED_toggle
    toggle LED_0
    goto Main
end
```

Figura 2. 11: Representación de etiquetas Basic.
Fuente: Autor

Mientras que Verle, M. (2010) sostiene que las etiquetas proporcionan la manera más sencilla de controlar diagramas ASM (flujograma), utilizados para denotar líneas particulares de un programa, para permitir que una instrucción de salto y subrutinas sean ejecutadas correctamente. Por lo general los compiladores de Basic pueden reconocer dichas etiquetas si al final de una

línea de programa va así ':'. En la figura 2.12, Verle, M. (2010) muestra como debe ir programada en MikroBasic.

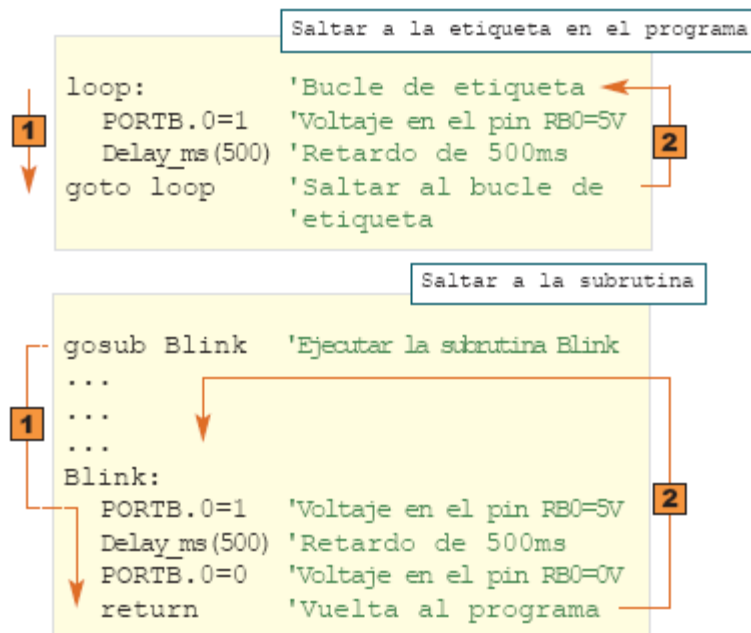


Figura 2. 12: Representación en MikroBasic de una etiqueta.
Fuente: Verle, M. (2010)

2.5.3. Constantes.

Las constantes son declaraciones que se les asigna un nombre cuyo valor es una constante. Es decir, si la constante minutos tiene el valor de 60 segundos en un minuto, que le puede escribir en cualquier posición del programa, minutos será interpretada por compilador como si hubiera sido escrito 60 s.

Mientras que Verle, M. (2010) indica que una constante es un número o cadena de caracteres que no podrá ser modificado en la compilación y corrida del programa. Estas son guardadas en la memoria ROM del uC que ocupan espacio posible de la memoria RAM.

En la figura 2.13 se muestra la sintaxis para declarar constantes y con una serie de ejemplos.

```

const nombre_de_constante [as type] = valor

const MINIMUM = -1000 ' Declarar la constante MINIMUM

const MAX as longint = 10000
const MIN = 1000 ' Compilador supone el tipo word
const SWITCH = "n" ' Compilador supone el tipo char

const T_MAX = 32.60 ' Declarar temperatura T_MAX
const T_MAX = 3.260E1 ' Otra forma de declarar la constante T_MAX

const Message_1 = "Press the START button" ' Mensaje 1 para LCD
const Message_2 = "Press the RIGHT button" ' Mensaje 2 para LCD
const Message_3 = "Press the LEFT button" ' Mensaje 3 para LCD

```

Figura 2. 13: Sintaxis y ejemplos en MikroBasic de una constante.
Fuente: Verle, M. (2010)

2.5.4. Variables.

Las variables sirven para el almacenamiento temporal de los datos y resultados de las diversas operaciones aritméticas y lógicas. Las variables se almacenan en las posiciones de la RAM de los microcontroladores, lo que significa que el número total de las variables que se pueden utilizar depende del tamaño de la memoria RAM.

En consecuencia para un microcontrolador PIC de 36 bytes, significaría que 22 bytes están reservados para las variables. En la figura 2.14 se muestra la sintaxis que se utiliza en Basic.

```

dim nombre_de_variable as type

dim i, j, k as byte 'Definir variables i, j, k
dim counter, temp as word 'Definir variables contador y temp

```

Figura 2. 14: Sintaxis y ejemplos en MikroBasic de una variable.
Fuente: Verle, M. (2010)

2.5.5. Símbolos.

Los símbolos (*symbols*) sirven para la asignación de los nuevos nombres de las variables y constantes. Los símbolos se introducen para la compatibilidad de los programas escritos para MikroBasic y no pueden ser utilizados para la introducción de las variables.

Mientras que Verle, M. (2010) sostiene que los símbolos deben ser declarados al inicio del módulo o programa, por debajo del nombre de módulo y la directiva 'include', tal como se ilustra en la figura 2.15.

```
symbol nombre_de_símbolo = código

symbol MAXALLOWED = 216      ' Símbolo MAXALLOWED para el valor numérico
symbol OUT = PORTA          ' Símbolo OUT para SFR
symbol MYDELAY = Delay_ms(762) ' Símbolo MYDELAY para llamada de procedimiento
dim cnt as byte              ' Variable cnt
main:
if cnt > MAXALLOWED then    ' Programa comprueba si cnt > 216
    cnt = 0                  ' Si lo es,
    OUT.1 = 0                ' los siguientes tres comandos
    MYDELAY                  ' se ejecutarán

end if
...                          ' Si no lo es, el programa continúa aquí
```

Figura 2. 15: Sintaxis y ejemplos en MikroBasic de símbolos.
Fuente: Verle, M. (2010)

CAPÍTULO 3: Comunicaciones Inalámbricas.

En el presente capítulo se describirá el medio de transmisión inalámbrico, que es el escogido para controlar al robot acuático, no se entrará en muchos detalles pero se fundamentará lo más relevante de las comunicaciones inalámbricas.

3.1. Introducción a Redes Inalámbricas.

El espectro radioeléctrico es un recurso natural limitado, con un fuerte contenido económico y estratégico. Este recurso es limitado por que implicaría realizar una buena administración del mismo para conseguir su máximo aprovechamiento y una utilización eficaz.

El espectro radioeléctrico es de dominio público y administrado exclusivamente por el Estado, en este caso por el Gobierno Ecuatoriano. Donde la administración recae al Ministerio de Telecomunicaciones (MINTEL), Consejo Nacional de Telecomunicaciones (CONATEL) y Superintendencia de Telecomunicaciones (SUPERTEL). En la cual la administración incluye, tareas de planificación, gestión y control. Por muchos motivos tanto técnicos, económicos, como estratégicos. Estas tareas tienen una fuerte implicación internacional, por ello dicha administración ha de hacerse de acuerdo con los Tratados y Acuerdos internacionales en los que Ecuador forma parte, tales como los establecidos en el marco de la Unión Internacional de Telecomunicaciones (UIT).

Mientras que Tanenbaum, A. S. (2003) sostiene que en el vacío las ondas electromagnéticas viajan a una misma velocidad, independientemente del valor de su frecuencia, la misma se la conoce como la velocidad de la luz 'c', es de aproximadamente 3×10^8 m/s o de 1 ft/ns. A manera de ejemplo, tenemos que tanto para cobre (Cu) y fibra óptica, la velocidad se reduce a $\frac{2}{3}c$ y dependiente de la frecuencia.

El espectro radioeléctrico se subdivide en bandas de frecuencias que incluyen las bandas de frecuencias para el desarrollo Industrial, Científico y Médico (ISM), y dentro de ellas el desarrollo de las tecnologías de acceso inalámbrico. En la tabla 3.1 se muestra las denominaciones de las bandas de Frecuencia del Espectro Radioeléctrico.

Tabla 3. 1: Clasificación de las Bandas de Frecuencia del Espectro Radioeléctrico

Banda	Rango de frecuencias	Rango de longitudes de onda en el espacio libre	Características de propagación	Uso típico
Frecuencias extremadamente bajas (ELF, <i>Extremely Low Frequency</i>)	30 a 300 Hz	10.000 a 1.000 km	GW	Líneas de potencia; se utilizan en algunos sistemas de control domésticos
Frecuencias de voz (VF, <i>Voice Frequency</i>)	300 a 3.000 Hz	1.000 a 100 km	GW	Se usan en los bucles de abonado de los sistemas de telefonía
Frecuencias muy bajas (VLF, <i>Very Low Frequency</i>)	3 a 30 kHz	100 a 10 km	GW con baja atenuación diurna y nocturna; alto nivel de ruido atmosférico	Navegación en alta mar; comunicaciones submarinas
Frecuencias bajas (LF, <i>Low Frequency</i>)	30 a 300 kHz	10 a 1 km	GW; ligeramente menos fiable que VLF; absorción diurna	Navegación en alta mar; radiolocalización para comunicaciones marinas
Frecuencias medias (MF, <i>Medium Frequency</i>)	300 a 3.000 kHz	1.000 a 100 m	GW y SW nocturna; baja atenuación nocturna, siendo alta la diurna; ruido atmosférico	Radio marítima; búsqueda de direcciones; radiodifusión AM
Frecuencias altas (HF, <i>High Frequency</i>)	3 a 30 MHz	100 a 10 m	SW; la calidad varía a lo largo del día, con las estaciones y la frecuencia	Radioaficionados; radiodifusión internacional; comunicaciones militares; navegación aérea de larga distancia y comunicaciones marítimas
Frecuencias muy altas (VHF, <i>Very High Frequency</i>)	30 a 300 MHz	10 a 1 m	LOS; dispersión (<i>scattering</i>) debido a la inversión de temperaturas; ruido cósmico	Televisión VHF; radiodifusión FM, comunicaciones AM en aviones; ayudas a la navegación de aviones
Frecuencias ultra altas (UHF, <i>Ultra High Frequency</i>)	300 a 3.000 MHz	100 a 10 cm	LOS; ruido cósmico	Televisión UHF; telefonía celular; radar; enlaces de microondas; sistemas de comunicación personal
Frecuencias super altas (SHF, <i>Super High Frequency</i>)	3 a 30 GHz	10 a 1 cm	LOS; la lluvia atenúa por encima de 10 GHz; atenuación atmosférica debido al vapor de agua y al oxígeno	Comunicaciones satelitales; radar; enlaces de microondas terrestres; bucles locales inalámbricos
Frecuencias extremadamente altas (EHF, <i>Extremely High Frequency</i>)	30 a 300 GHz	10 a 1 mm	LOS; atenuación atmosférica debido al vapor de agua y al oxígeno	Experimental; bucles locales inalámbricos
Infrarrojos	300 GHz a 400 THz	1 a 770 nm	LOS	LAN infrarrojas; aplicaciones de electrónica de consumo
Luz visible	400 a 900 THz	700 a 330 nm	LOS	Comunicaciones ópticas

Fuente: Stallings, W. (2004)

Elaborado: Autor

En la tabla 3.2 se especifica la lista de las denominaciones de las bandas en el rango conocido como de las microondas.

Tabla 3. 2: Lista de bandas disponibles en la Clasificación de las Bandas de Frecuencia del Espectro Radioeléctrico

Banda S	2- 4 GHz
Banda C	4-8 GHz
Banda X	8 - 12 GHz
Banda Ku	12 - 18 GHz
Banda Ku	18 - 27 GHz
Banda Ka	27 - 40 GHz

Elaborado: Autor

En la tabla 3.3 se muestran las tres frecuencias en que operan las redes inalámbricas.

Tabla 3. 3: Rango de frecuencias para enlaces de redes inalámbricas

	902 - 928 MHz
Banda de 2.4 GHz	2.4 - 2.4835 GHz
Banda de 5 GHz	5.725 - 5.875 GHz

Elaborado: Autor

Las tecnologías de acceso inalámbrico permiten disponer en la actualidad, de sistemas de alta velocidad para ser utilizadas en diferentes modos de acceso, es decir, ya sea en las redes privadas con servicios internos de una misma entidad, y/o en redes públicas con disponibilidad de servicio al público en general o sectores específicos de carácter comercial.

Una red de área local inalámbrica o también “Wireless LAN”- (de aquí en adelante RLAN), puede definirse como una red de alcance local que tiene como medio de transmisión el aire.

3.2. Propagación Inalámbrica.

Stallings, W. (2004) sostiene que en la transmisión radiada por antenas siguen tres tipos de trayectorias posibles, que son:

a. Superficial.

Stallings, W. (2004) sostiene que mediante la propagación superficial de señales (por debajo de 2 MHz), estas permiten comunicarse a grandes distancias, superando así la línea de horizonte visual, tal como se muestra en la figura 3.1.

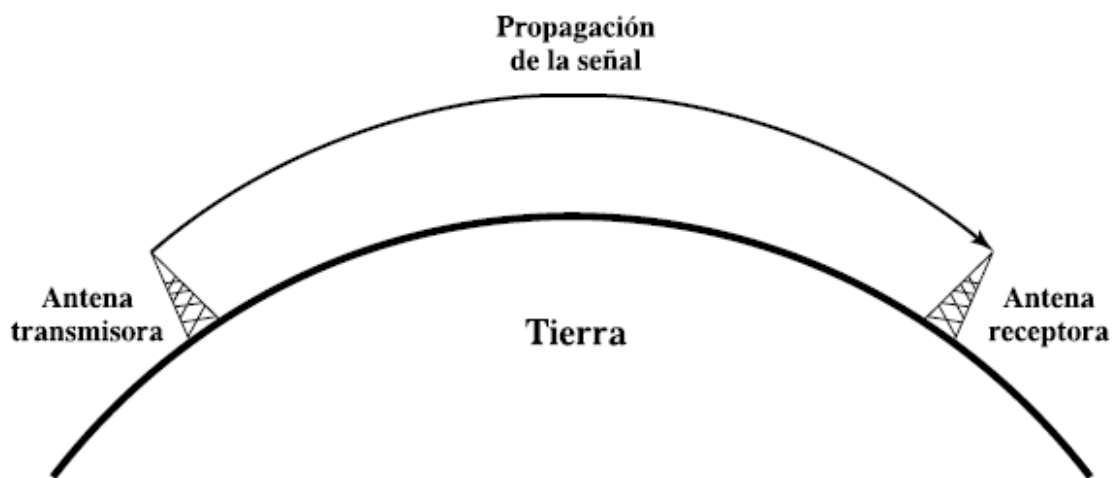


Figura 3. 1: Modo de propagación superficial.
Fuente: Stallings, W. (2004)

b. Aérea.

Stallings, W. (2004) indica que la propagación aérea de señales, es solo utilizada por: radios comerciales (por ejemplo: la radio BBC), radioaficionados. Siendo este tipo de propagación refractario, es decir, que se propagan mediante una serie de saltos, lo que permite transmitir a miles de kilómetros del transmisor, tal como se muestra en la figura 3.2.

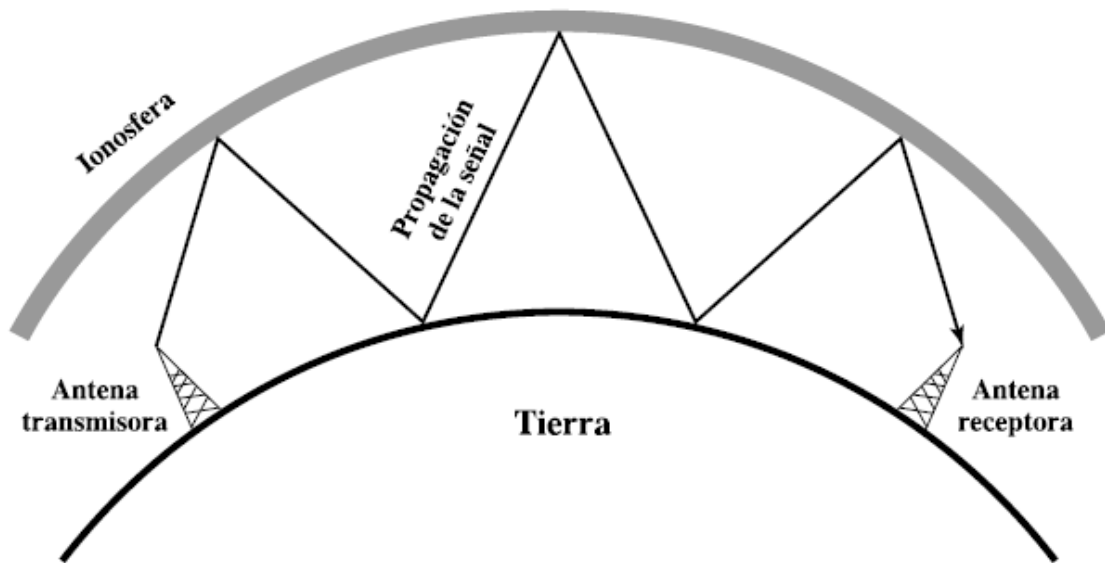


Figura 3. 2: Modo de propagación aérea.
Fuente: Stallings, W. (2004)

c. Visual.

Stallings, W. (2004) manifiesta que cuando no se puede transmitir información mediante propagación superficial o área, debe de existir línea de vista, tal como se muestra en la figura 3.3.

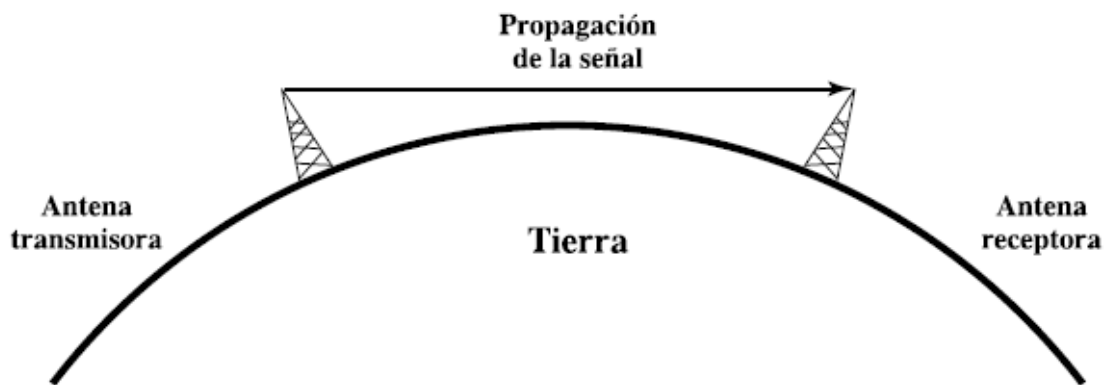


Figura 3. 3: Modo de propagación mediante línea de vista.
Fuente: Stallings, W. (2004)

Para este tipo de propagación se debe hacer un análisis de como transmitir sin obstáculos, es decir, que entre dos antenas debe existir línea de vista, y esto se logra mediante las siguientes ecuaciones:

$$d = 3.57\sqrt{Kh}$$

La ecuación anterior sirve para calcular la distancia total entre dos antenas con alturas diferentes.

$$d = 3.57(\sqrt{Kh_1} + \sqrt{Kh_2})$$

Mientras que en esta última ecuación, se encontraría la altura de una de las dos antenas.

3.3. Estándares de Redes Inalámbricas 802.11.

El IEEE, el ETSI y una alianza de tecnología (HomeRF) fomentan los estándares de redes inalámbricas. En la familia de IEEE 802.11 de RLANs, tres estándares merecen atención individual, los demás se mencionan brevemente.

Mientras que Tanenbaum, A. S. (2003) indica que los protocolos utilizados en 802, entre ellas Ethernet, guardan similitudes de estructura. El estándar 802.11 fue aprobado en julio de 1997, siendo el primer estándar para ese entonces similar al Ethernet (alámbrico), con la diferencia que empleaba transmisión inalámbrica cuya banda de frecuencia es 2.4 GHz. Su velocidad de transmisión era entre 1 y 2 Mbps, a través de saltos de frecuencias (FHSS) o de secuencia directa (DSSS).

El estándar principal es 802.11b, o WiFi (Wireless Fidelity). El segundo es 802.11a, que da mayor razón de datos a una frecuencia más alta y sin tanta interferencia, el tercero es 802.11g, otros estándares son HIPERLAN/1 y HIPERLAN/2. Mientras que Valdivia M., C. (2014) describe los estándares de las redes inalámbricas:

a. Estándar IEEE 802.11a.

Opera en la banda de 5 GHz y puede comunicarse a 54 Mbps, emplea multiplexación de frecuencia de división ortogonal (OFDM), que proporciona corrección en transmisión digital y almacenamiento de

consulta de memoria digital, permite razones de datos hasta 72 Mbps usando 8 canales no solapados.

b. Estándar IEEE 802.11b.

El 802.11b es el estándar más popular en la familia 802.11x, siendo hasta el momento la que ha logrado amplia aceptación en el mercado para redes inalámbricas. El 802.11b está basado sobre la versión DSSS de 802.11 ya que es más fácil de implementar que OFDM, en la banda de 2.4-GHz.

c. Estándar IEEE 802.11g.

Aprobado en noviembre de 2001, opera en la banda 2.4 GHz con velocidades de hasta 54 Mbps, siendo compatible con 802.11b y de mejor alcance que la 802.11.a, utiliza modulación de DSSS para velocidades hasta 11 Mbps, para 54 Mbps, usa los más eficientes tipos de modulación del OFDM.

d. Estándar IEEE 802.1n.

Aumenta la velocidad de las redes inalámbricas entre 100 y 300 Mbps. se espera que alcance los 500 Mbps.

e. Estándar IEEE 802.11e.

Dirigido a proporcionar capacidades de calidad de servicio (QoS, *Quality of Service*) permite comunicación de voz confiable en sistemas de 802.11b. Adicionalmente este estándar ha mejorado su seguridad y los mecanismos de autenticación.

f. Estándar IEEE 802.11f.

Dirigido a desarrollar las prácticas, recomendado por IAPP (*Inter-Access Point Protocol*) al lograr interoperabilidad de AP de fabricantes diferentes.

g. Estándar IEEE 802.11h.

Dirigido a mejorar el nivel físico de alta velocidad del 802.11a en la banda de 5 GHz para hacer productos del IEEE 802.11a que complementan las normas Europeas.

h. Estándar IEEE 802.11i.

Mejora el nivel MAC de 802.11 incrementando la seguridad y los mecanismos de autenticación.

3.4. Estándar Home RF

Es una tecnología dirigida a las redes inalámbricas del hogar, está conformada a partir del estándar 802.11 FHSS pero mejorado, con el objetivo de soportar las necesidades únicas del consumidor, usa el protocolo de acceso inalámbrico compartido (*Shared Wireless Access Protocol- SWAP*).

Pazmiño O., L. A. & Sánchez G., R. A. (2010) cita en su trabajo de titulación que Home RF es una tecnología de comunicación por radio frecuencia que está basada en el protocolo de acceso compartido SWAP y cuyo grupo de trabajo es *Home RF Working Group* (HHRFWG), el cual tiene por misión, encaminar a este estándar hacia un buen desarrollo tecnológico y lograr que sus elementos sean plenamente compatibles e integrables en una misma instalación.

Uno de los aportes más significativos de SWAP es su soporte para la comunicación de voz de alta fidelidad. Adicionalmente la especificación incorpora el estándar DECT (*Digital Enhanced Cordless Telephony*), este permite a los teléfonos inalámbricos usar la misma infraestructura de la red de la casa como computadoras y electrodomésticos.

En la tabla 3.4 se muestran cada una de las características del estándar Home RF.

Tabla 3. 4: Características del estándar Home RF.

Características	Home RF 1.0	Home RF 2.0
Banda de frecuencia	2.4 GHz. ISM	2.4 GHz. ISM
Método de acceso al canal	1 MHz de ancho de frecuencia.	1 MHz a 5 MHz de ancho de frecuencia.
Máximo ancho de banda para transmisión de datos.	1.6 Mbps.	10 Mbps.
Ancho de banda utilizable por el usuario en presencia de interferencias causadas por otro tipo de dispositivos que utilizan radiofrecuencia.	650 Kbps.	5 Mbps.
Radio de acción en interiores.	Hasta 50 m.	Hasta 50 m.
Potencia transmitida.	100 mW.	100 mW y 500 mW.
Flujos de datos con prioridad sobre las demás funciones para sesiones de audio y video.	No.	Si, hasta 8 simultáneamente.
Soporte para voz.	4 conjuntos basados en DECT.	4 conjuntos basados en DECT, ampliables hasta 8.
Roaming.	No.	Si.
Encriptación para soporte MAC.	40 bits.	128 bits.
Consumo mínimo de potencia para protocolo TCP/IP.	< 10 mW	< 10 mW.
Factores de forma OEM.	PC Card, USB, Módulo OEM, Compact Flash y Ethernet.	PC Card, USB, Módulo OEM, Compact Flash y Ethernet.

Fuente: Pazmiño O., L. A. & Sánchez G., R. A. (2010)

HomeRF ha sido diseñado para dar facilidad de uso con un bajo precio, proporcionando rangos hasta 50 metros con velocidades máximas de 10 Mbps, usa la banda de frecuencia de 2.4 GHz. Con la adopción amplia de la industria del 802.11b, los dispositivos de HomeRF no han podido vencer el mercado.

3.5. Estándar HIPERLAN/1 e HIPERLAN/2.

El ETSI planteó el estándar HIPERLAN (*High-Performance Radio Local Area Network*) en 1992 para dirigir la necesidad de la comunicación inalámbrica de corto alcance a alta velocidad. La versión primera normalmente se refiere al

HIPERLAN/1, está basada en estándares Ethernet, con su transmisión de radio basado en GSM, usa la banda de frecuencia de 5 GHz. El rango de operación y ancho de banda son difíciles de determinar porque el HIPERLAN/1 no progresó comercialmente, según la especificación el HIPERLAN tiene razones de datos de cerca de 23.5 Mbps.

Mientras que HIPERLAN/2 soporta dos modos de operación básicos: Modo Centralizado (CM) y Modo Directo (DM). El primero es aplicable a topologías de redes celulares, en la cual cada celda de radio es controlada por un Access Point (AP) que abarque el área geográfica. Es decir, que el modo centralizado, permite transmisiones entre sí de teléfonos móviles entre o con el núcleo de la red, todo esto a través de un AP, siendo esto muy utilizado en aplicaciones comerciales de interior y de exterior donde el área que se ha de cubrir es más grande que una celda de radio.

El segundo (Modo Directo) es solo aplicable para redes en modo Ad-Hoc, es decir, para hogares privados y donde toda el área que sirve se cubre por una celda de radio.

3.6. Redes Inalámbricas con tecnología WiFi.

WiFi también conocida como Red Inalámbrica, debido a que su capacidad de transmisión y alcance para acceder a la nube sin necesidad de cableado. Inicialmente WiFi se utilizaba solo para los aparatos cuya tecnología pertenecía al estándar IEEE 802.11b, siendo de aceptación prácticamente universal, que funciona en la banda de frecuencias de 2,4 GHz que a su vez está subdividida en 14 canales, permitiendo la transmisión de datos a una velocidad de hasta 11Mbps.

Con el fin de evitar confusiones en la compatibilidad de los aparatos y la interoperabilidad de las redes, el término WiFi se extendió a todos los aparatos

provistos con tecnología 802.11:, 802.11b, 802.11g. En el Anexo 4 se muestra una tabla comparativa entre 802.11b y 802.11g.

En este proyecto la comunicación inalámbrica utilizada es ISM (**Industrial, Scientific and Medical**) son bandas reservadas internacionalmente para uso no comercial de El uso de estas radiofrecuencia electromagnética en áreas industrial, científica y médica. En la actualidad estas bandas han sido popularizadas por su uso en comunicaciones WLAN (*WI-FI*) o WPAN (Bluetooth).

Bandas de frecuencia está abierto a todo el mundo sin necesidad de licencia, respetando las regulaciones que limitan los niveles de potencia transmitida. Este hecho fuerza a que este tipo de comunicaciones tengan cierta tolerancia frente a errores y que utilicen mecanismos de protección contra interferencias, como técnicas de ensanchado de espectro.

Los XBee pueden comunicarse en arquitecturas punto a punto, punto a multipunto o en una red mesh (malla). La elección del módulo XBee correcto pasa por escoger el tipo de antena (chip, alambre o conector SMA) y la potencia de transmisión (2mW para 300 pies o 60mW para hasta 100 metros) IEEE® 802.15.4 OEM RF Modules by MaxStream, Inc. (asegura que no se pierde ni se retransmite) operan en frecuencias: ISM 2,4 GHZ

La cual es utilizada con: Antenas XBEE (recepción la que se encuentra en el barco) (Trasmisión la que está conectada al computador).

CAPÍTULO 4: DESARROLLO EXPERIMENTAL A ESCALA

4.1. Hardware del Prototipo a escala.

El prototipo electrónico a escala que se estudia en detalle, nos permite mantener el control de su sistema mediante un acoplamiento remoto por antena XBEE, utilizando una interfaz de comunicación serial UART para el envío de los comandos de movimiento desde el computador.

Se ha desarrollado una tabla de interconexión (véase la tabla 4.1) entre los dispositivos y la tarjeta de control con microcontrolador PIC16F886 (véase la figura 4.1), así como también sus interfaces de relación como Puente H (véase la figura 4.2) y sensores de distancia ultrasónicos (véase la figura 4.3).

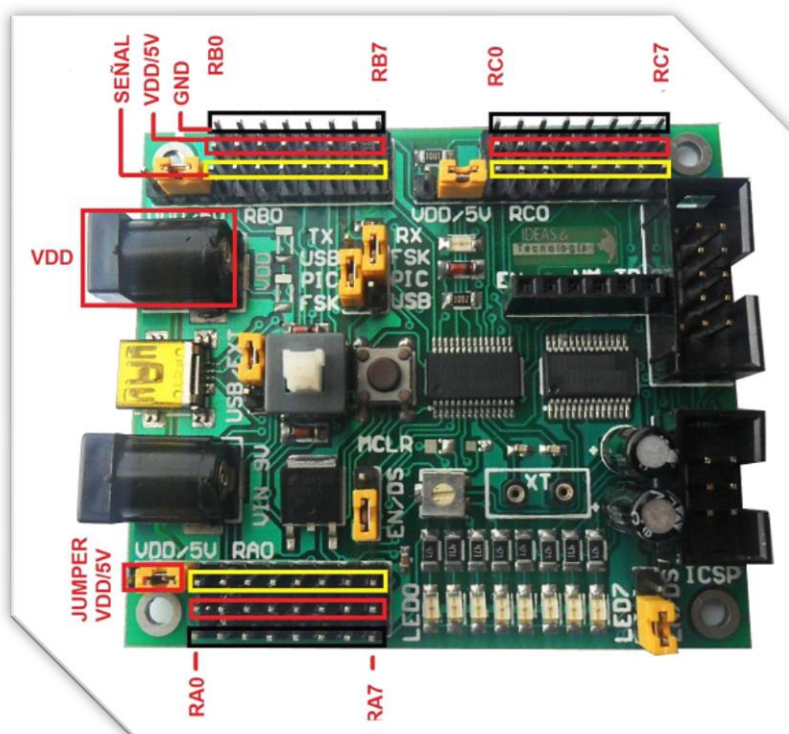


Figura 4. 1: Tarjeta de control con PIC16F886.
Fuente: Ideas&Tecnologías.

Los dispositivos electrónicos representados por las figuras 4.1 y 4.2 sirven para diferentes aplicaciones, las mismas que dependerán de los tipos de proyectos a desarrollar.

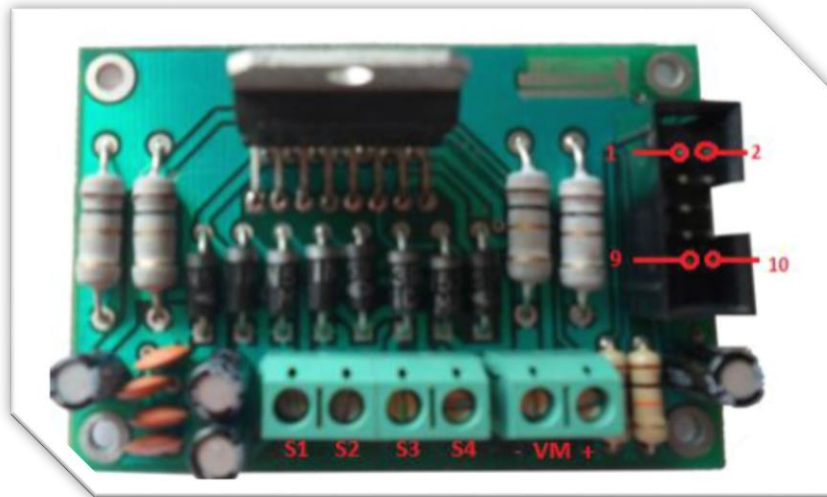


Figura 4. 2: Tarjeta del Puente H.
Fuente: Ideas&Tecnologías.

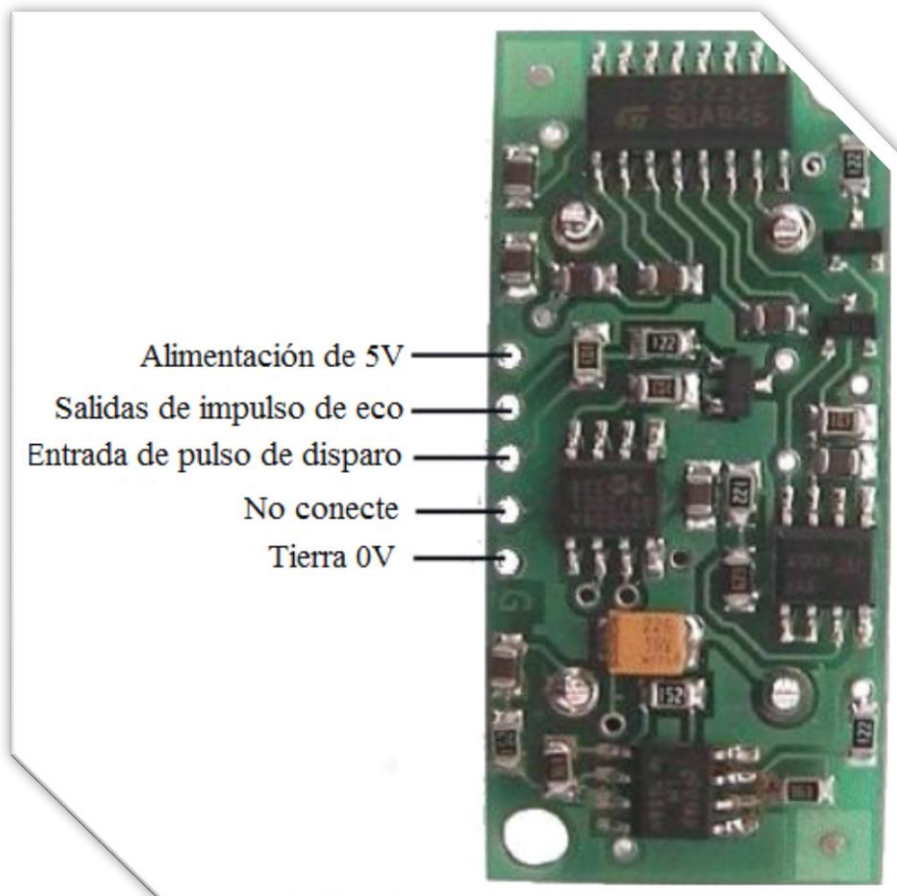


Figura 4. 3: Tarjeta del sensor de distancia ultrasónico.
Fuente: robot-electronics.

Tabla 4. 1: Interconexión entre dispositivos.

TARJETA CONTROLADORA	PUENTE H	OBSERVACIÓN
PORTA2	PIN1	CONTROL DIRECCION MOTOR1
PORTA4	PIN3	CONTROL DIRECCION MOTOR1
PORTA3	PIN5	CONTROL DIRECCION MOTOR2
PORTA5	PIN7	CONTROL DIRECCION MOTOR2

En la tabla 4.2 se muestra las características principales del módulo de entrenamiento:

Tabla 4. 2: Características de la tarjeta de control.

Características
UART, One Wire, SPI e I2C.
Comunicación ONE WIRE y USART.
Rx y Tx (FSK, ASK y QPSK).
Potenciómetro integrado.
10 entradas analógicas.
24 entradas y salidas digitales.
8 leds indicadores de salidas digitales.
Control para 2 motores DC (Dirección y Velocidad).
Programación ICSP in circuit.
Reset manual.
Switch de ON/OFF.
Led indicador de power.
Regulador integrado.

El Puente H presenta la siguiente configuración de pines para completar el enlace de control y fuerza, que se muestra en la tabla 4.3.

Tabla 4. 3: Configuración del puente H.

PUENTE H	PINES
Entrada de control 1	PIN1
Señal habilitadora motor 1	PIN2
Entrada de control 2	PIN3
Señal habilitadora motor 2	PIN4
Entrada de control 3	PIN5
No conectar	PIN6
Entrada de control 4	PIN7
No conectar	PIN8
VCC=5 VDC	PIN9
GND	PIN10

Ahora, la Interconexión de la Tarjeta de control con el Módulo Antena XBEE, este utiliza la comunicación serial Uart y actúa como esclavo, actualmente conectado a la tarjeta de control y recibe los datos enviados desde la aplicación. En la tabla 4.4 se muestra la configuración de conexión entre los pines de la antena XBEE y la tarjeta de control.

Tabla 4. 4: Configuración pines entre tarjeta de control y antena XBEE.

TARJETA DE CONTROL	XBEE
PORTC.6	RX
PORTC.7	TX

Mientras que la interconexión Tarjeta de control y Sensores Ultrasónicos, tiene como propósito utilizar estos sensores de distancia ultrasónicos, con el único afán de simular situaciones en la que un robot permita detectar obstáculos, que impidan el paso del vehículo, y dependiendo de los datos obtenidos, el móvil tomará decisiones y control del movimiento para evitar colisiones, todo ello es realizado en ambiente de laboratorio con el propósito del estudio de datos y la forma como se puede incorporar esta interface al microcontrolador.

En la tabla 4.5 se muestra la configuración entre los pines de la tarjeta de control y del sensor ultrasónico.

Tabla 4. 5: Configuración entre tarjeta de control y sensor ultrasónico.

TARJETA DE CONTROL	SENSOR ULTRASÓNICO FRENTE
PORTC.5	TRIGGER
PORTC.4	ECHO
TARJETA DE CONTROL	SENSOR ULTRASÓNICO ATRÁS
PORTB.4	TRIGGER
PORTB.5	ECHO

En la figura 4.4 se muestra el diagrama de bloques de la interconexión del hardware utilizado para el desarrollo del prototipo “robot acuático radiocontrolado”, es decir, es la interconexión entre la tarjeta de control PIC16F886 hacia el puente H y como al módulo de comunicación inalámbrica con antena XBEE, este último se comunica a otra antena XBEE instalada en un computador fijo o portátil.

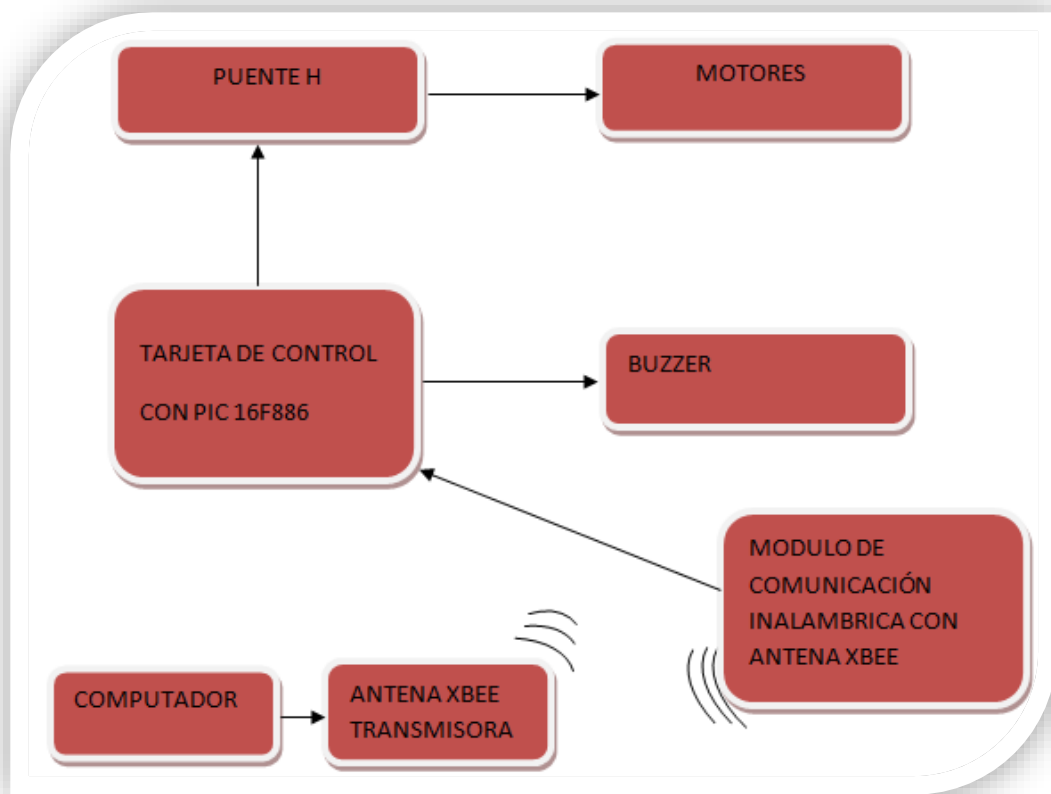


Figura 4. 4: Diagrama de bloques de interconexión entre dispositivos electrónicos del prototipo.

Fuente: Autor.

Para el presente trabajo de titulación utilizaremos el programador P.PICI&T04 (véase la figura 4.5), el mismo es muy importante, ya que nos permite cargar o grabar el archivo “.HEX” en el microcontrolador.

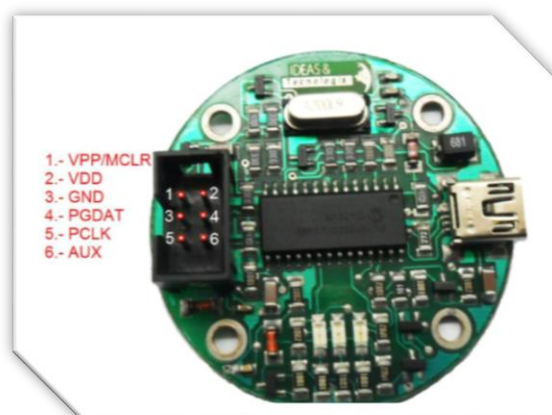


Figura 4. 5: Programador P.PICI&T04.

Fuente: El Autor.

4.2. Software: Programación del Hardware

Independientemente de la aplicación desarrollada (hardware) se deberá programar al módulo de control PIC16F886, y esa programación dependerá de la necesidad que tengamos. El lenguaje de programación de alto nivel elegido para este prototipo es Mikrobasic, posteriormente a la programación del hardware, se cargará el programa desarrollado al módulo de control PIC16F886.

En la figura 4.6 se ilustra el proceso de la programación del microcontrolador.

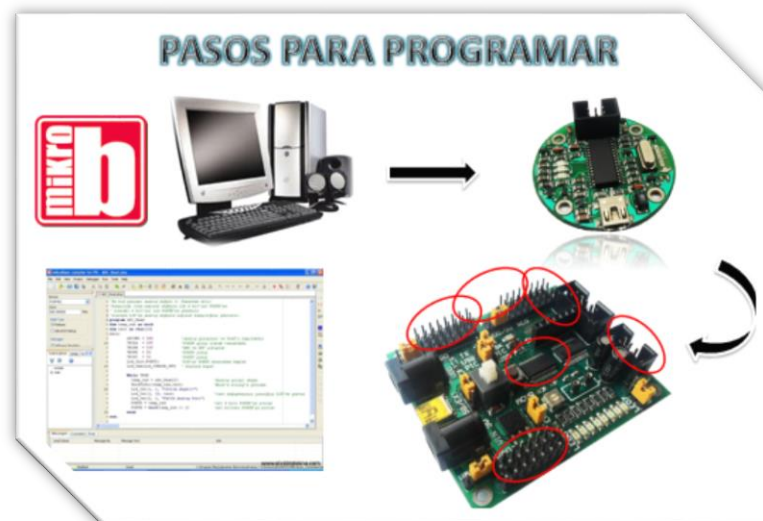


Figura 4. 6: Pasos para la programación.
Fuente: El Autor.

Para la programación de la tarjeta controladora PIC16F886, debemos tener en cuenta que componentes y cuales puertos del PIC en mención tendríamos que emplear para una correcta programación del PIC. En la figura 4.7 se muestra la descripción de los pines del PIC16F886.

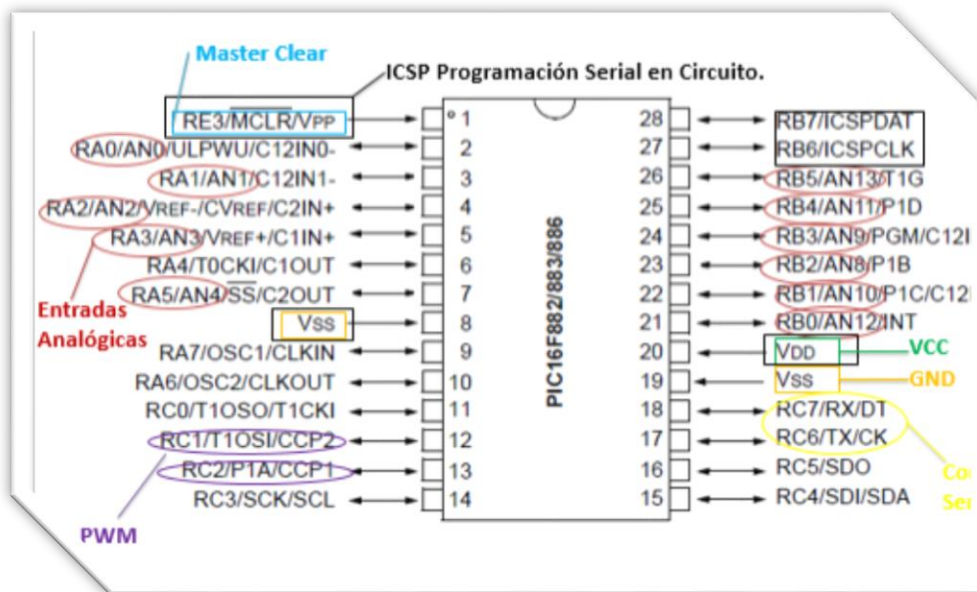


Figura 4. 7: Descripción de pines del PIC16F886.
Fuente: El Autor.

Posteriormente se deberá cargar el programa una vez compilado y generado el archivo *.hex, tal como se muestra en la figura 4.8.

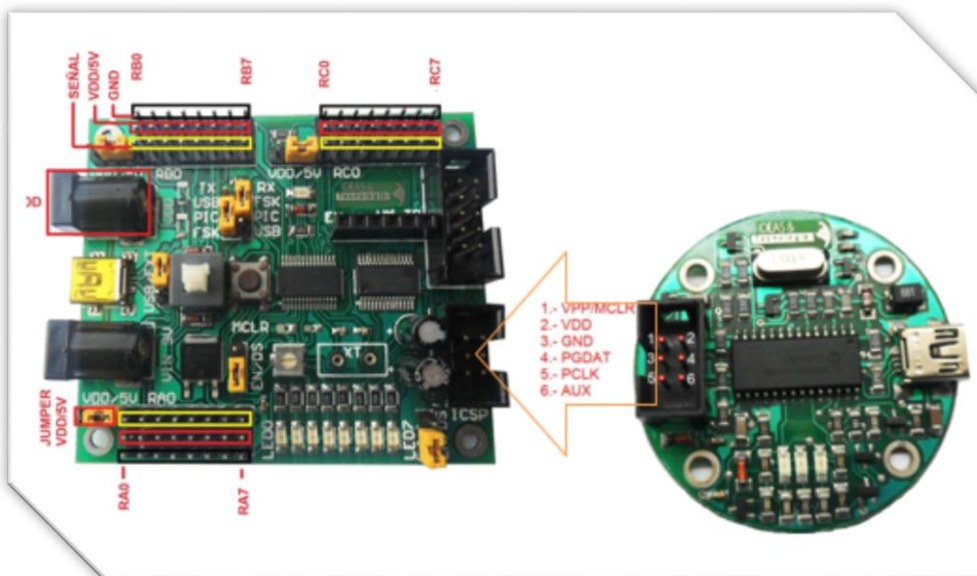


Figura 4. 8: Conexión entre el programador y módulo de entrenamiento.
Fuente: El Autor.

4.3. Programación del Prototipo en MikroBasic.

Antes de programar en MikroBasic el prototipo del robot acuático o barco radiocontrolado, se describirá los diagramas ASM para la programación del módulo controlador PIC16F886, la misma que se encargará de comunicarse con la antena XBEE y el puente H que permite el movimiento de los motores.

En la figura 4.9 se muestra el diagrama ASM, que de acuerdo al mismo será programada para la subrutina comunicación UART, obviamente debe estar configurado las variables, posterior preguntará si hay comunicación UART, si es “Si” o “Verdadera” el sistema se podrá mover hacia “adelante”, “parar”, “atrás”, “izquierda” y “derecha”.

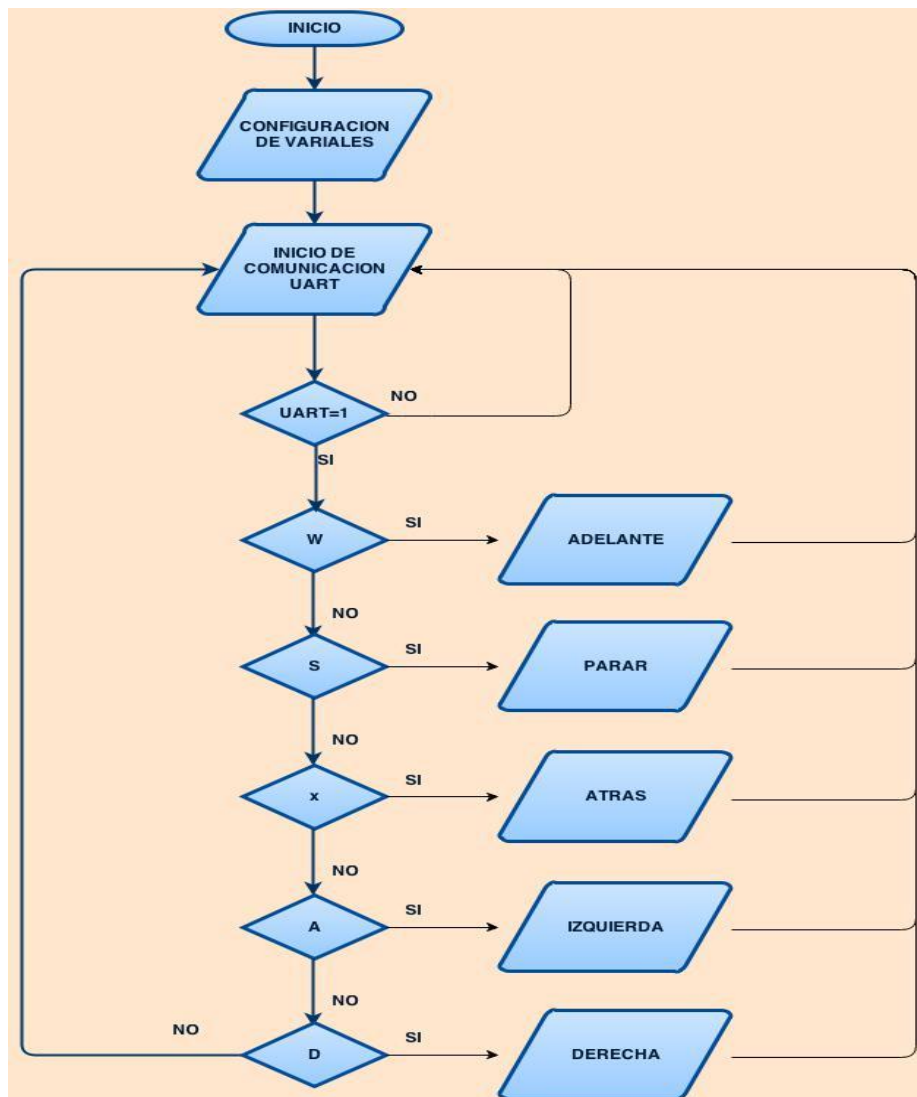


Figura 4. 9: Diagrama ASM para configuración UART.
Fuente: El Autor.

Posteriormente, mostramos en las figuras 4.10 y 4.11 los diagramas ASM de las subrutinas de “avanzar adelante” y “avanzar atrás” respectivamente.

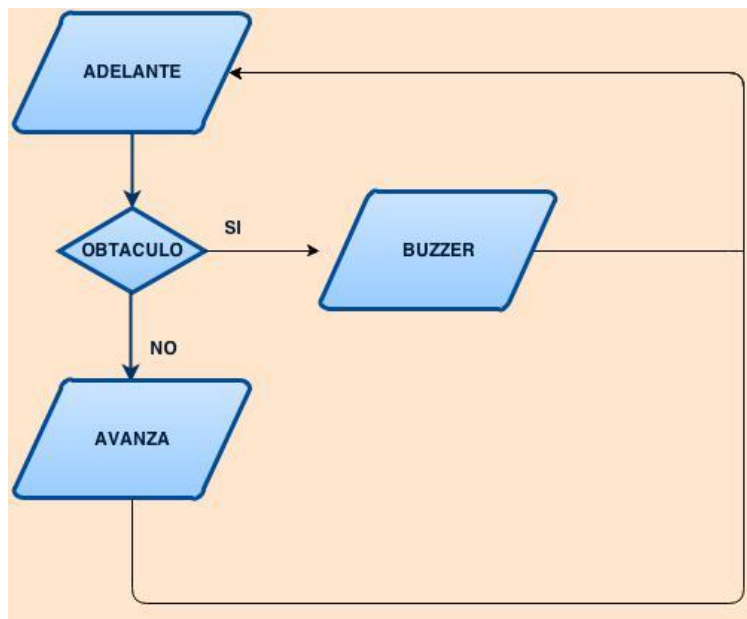


Figura 4. 10: Diagrama ASM para subrutina avanzar adelante.
Fuente: El Autor.

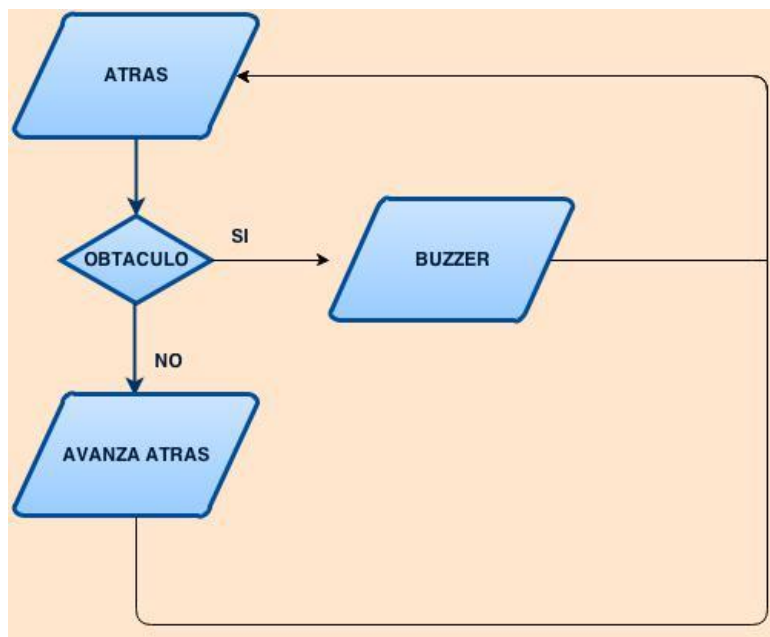


Figura 4. 11: Diagrama ASM para subrutina avanzar atrás.
Fuente: El Autor.

Para la comunicación inalámbrica entre el computador portátil y el módulo o antena XBEE, se utilizará una interfaz de comunicación denominada AccessPort, la misma es una interfaz que se instalará en un computador, sin esta aplicación no se podrá comunicar al prototipo. En la figura 4.12 se muestra la ventana de la interfaz AccessPort.

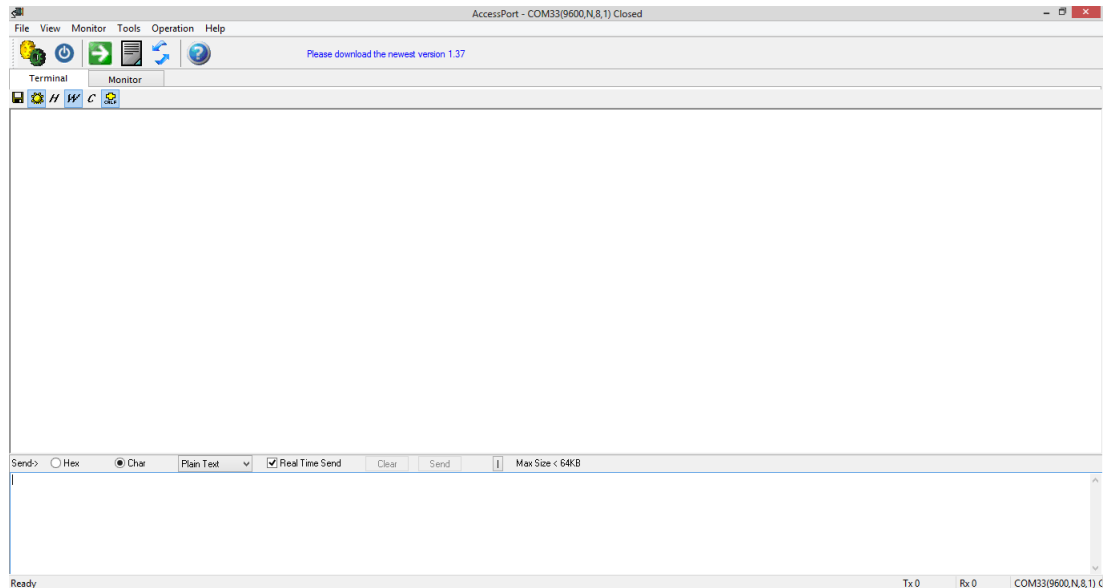


Figura 4. 12: Interfaz del computador AccessPort.
Fuente: El Autor.

Después de describir los diagramas ASM, se muestra a continuación las figuras 4.13, 4.14 y 4.15 como queda el robot acuático a escala.

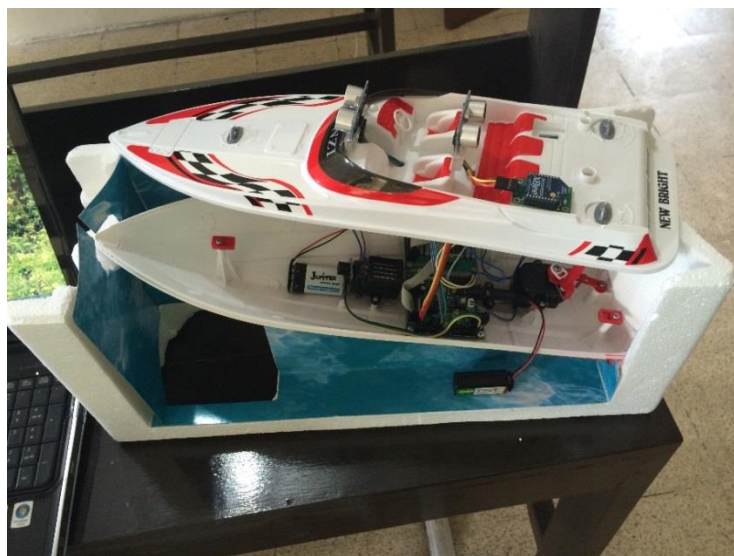


Figura 4. 13: Vista Superior del Robot a Escala.
Fuente: El Autor.

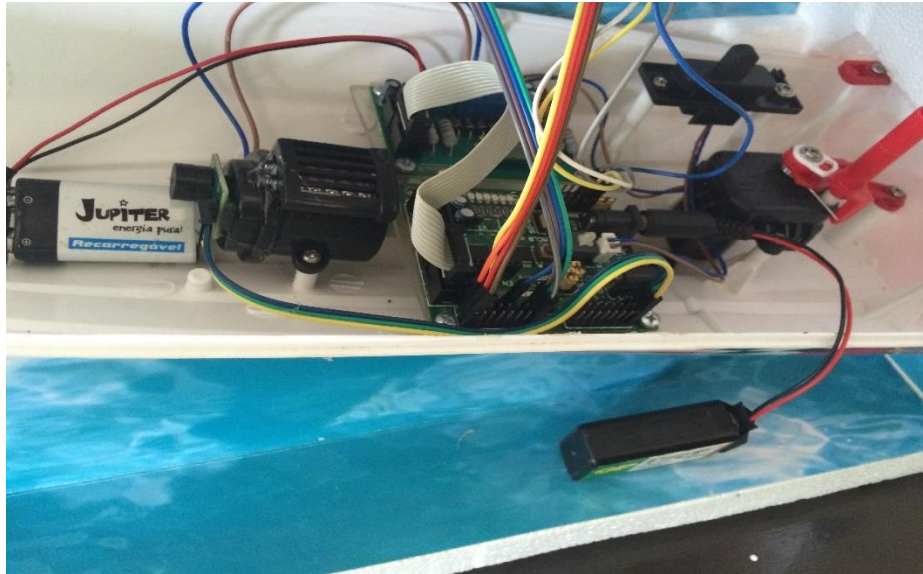


Figura 4. 14: Vista Interior de tarjetas electrónicas del robot a escala.
Fuente: El Autor.

Después de describir los diagramas ASM, se muestra a continuación la programación en MikroBasic.

CÓDIGO FUENTE EN MIKROBASIC PRO FOR PIC

```
program robot_barco_radiocontrolado
```

```
SYMBOL DIR1= PORTA.2 'CONTROL DIRECCION MOTOR1
SYMBOL NDIR1=PORTA.4 'CONTROL DIRECCION MOTOR1
```

```
SYMBOL DIR2= PORTA.3 'CONTROL DIRECCION MOTOR2
SYMBOL NDIR2=PORTA.5 'CONTROL DIRECCION MOTOR2
```

```
.....
' ULTRASONICO 1          'SENSOR FRENTE
SYMBOL DISPARO1 = PORTC.5 'PIN TRIGGER
SYMBOL ECO1 = PORTC.4    'PIN ECHO
.....
```

```
.....
' ULTRASONICO 2          'SENSOR ATRAS
SYMBOL DISPARO2 = PORTA.6 'PIN TRIGGER
SYMBOL ECO2 = PORTA.1    'PIN ECHO
.....
```

```
' declaracion de variables
```

dim distancia1, distancia2, distancia3,distancia4 as word
dim delante,modo,dato_recep,dato as byte

SUB PROCEDURE SET_MOTOR(DIM MOTOR2,MOTOR1 AS INTEGER)

***** MOTOR 1*****

IF(MOTOR1<0)THEN
MOTOR1=-MOTOR1
DIR1=0
NDIR1=1

ELSE
DIR1=1
NDIR1=0

END IF

***** MOTOR 2*****

IF(MOTOR2<0)THEN
MOTOR2=-MOTOR2
DIR2=0
NDIR2=1

ELSE
DIR2=1
NDIR2=0

END IF

PWM1_Set_Duty(MOTOR1)

PWM2_Set_Duty(MOTOR2)

END SUB

sub function ultrasonico1() as word

dim flanco as byte

dim distan,tiempo as word

flanco = 0

disparo1 = 1

Delay_us(50)

disparo1 = 0

DO

IF(eco1 = 1)THEN

TMR0 = 0

flanco = 1

END IF

LOOP UNTIL(flanco = 1)

DO

IF(eco1 = 0)THEN

tiempo = TMR0

flanco = 0

END IF

if(TMR0>254)THEN

```

        FLANCO=0
    END IF
LOOP UNTIL( FLANCO = 0 )

distan = (tiempo*128)/58
Delay_ms (50)
result = distan
end sub
!*****
sub function ultrasonico2( ) as word
    dim flanco as byte
    dim distan,tiempo as word

    flanco = 0

    disparo2 = 1
    Delay_us(50)
    disparo2 = 0

    DO
        IF( eco2 = 1 )THEN
            TMR0 = 0
            flanco = 1
            END IF
        LOOP UNTIL( flanco = 1 )

    DO
        IF( eco2 = 0 )THEN
            tiempo = TMR0
            flanco = 0
            END IF
            if(TMR0>254)THEN
                FLANCO=0
            END IF
        LOOP UNTIL( FLANCO = 0 )

    distan = (tiempo*128)/58
    Delay_ms (50)
    result = distan
end sub

main:
    OSCCON = 0X65

    OPTION_REG = %10000110
' REGISTRO PORTA
    TRISA = %10000011
    PORTA = %00000000

```

```

' REGISTRO PORTB
  TRISB = %00000000 'PIN RB1,RB2 ENTRADAS PARA SENSAR
CORRIENTE DE MOTORES
  PORTB = %00000000

' REGISTRO PORTC
  TRISC = %10010001
  PORTC = %00000000
' REGISTRO ANSEL
  ANSEL = %00000000 ' RE1-RE2 ANALOGICO, RESTO DIGITALES
  ANSELH = %00000000
  UART1_Init(9600)
' CONFIGURO EL PWM A UNA FRECUENCIA DE 1000HZ
  PWM1_Init(1000)
  PWM2_Init (1000)
  PWM1_Start ()
  PWM2_Start ()
  PWM1_Set_Duty (0)
  PWM2_Set_Duty (0)
  SET_MOTOR(255,255)
  Delay_ms(500)
  SET_MOTOR(0,0)
  dato=0
  dato_recep=0
  PORTB=0XFF
  Delay_ms(500)
  PORTB=0X00
WHILE(1)
  if (UART1_Data_Ready() = 1) then
    dato_recep = UART1_Read()
    dato= dato_recep
  end if

  if dato="W" then 'ADELANTE
    SET_MOTOR(255,255)
    PORTB=0XFF
  end if
  if dato="X" then 'ATRAS
    SET_MOTOR(-255,-255)
    PORTB=0X00
  end if
  if dato="S" then 'PARAR
    SET_MOTOR(0,0)
  end if
  if dato="A" then 'IZQUIERDA
    SET_MOTOR(0,0)
  end if

```



```
if dato="D" then 'DERECHA
  SET_MOTOR(0,0)
end if
```

```
WEND
end.
```

CAPÍTULO 5: CONCLUSIONES Y RECOMENDACIONES.

5.1. Conclusiones.

- Mediante el estudio realizado en el Estado del Arte de los Sistemas Microcontroladores PIC, se pudo fundamentar los principios básicos, características, familia de los PIC's y comprender la importancia de los microcontroladores a través de múltiples aplicaciones.
- Mediante la fundamentación teórica de las Comunicaciones Inalámbricas, se ha elegido como medio de transmisión, para la comunicación entre el computador y el robot acuático o barco radiocontrolado, a través de XBEE. Aunque las comunicaciones inalámbricas son aplicadas en diversidades de redes WLAN privadas o públicas, también son utilizados como medio de transmisión en aplicaciones a escala.
- La parte experimental nos permitió validar correctamente la aplicación específica de controlar el robot acuático. Se comprobó que la programación en MikroBasic permite transmitir y recibir datos al módulo control PIC16F886 de manera correcta.

5.2. Recomendaciones.

- Adquirir los módulos o tarjetas de control que permiten desarrollar aplicaciones prácticas de microcontroladores, así como también otros dispositivos o periféricos, todo esto en beneficio de los estudiantes de las Carreras de Ingeniería en Telecomunicaciones y Electrónica en Control y Automatismo.

- Adquirir las licencias originales de las plataformas de programación y simulación, es decir, MikroBasic y Proteus respectivamente, para disponer de librerías completas y evitar posibles sanciones por tener las plataformas pirateadas o hackeadas.

- Promover en los alumnos la investigación semilla, a través de cursos, talleres o seminarios de capacitación constante, ya sea en microcontroladores u otra tecnología, esto motivaría a los estudiantes de la Carrera de Ingeniería en Telecomunicaciones a proponer temas de trabajo de titulación o investigativos.

REFERENCIAS BIBLIOGRÁFICAS

Ángulo U., J. M. & Ángulo M., I. (2003). *Microcontroladores <<PIC>>. Diseño Práctico de Aplicaciones. Primera Parte: El PIC16F84. Lenguajes PBASIC y Ensamblador*. Tercera Edición. McGraw-Hill Interamericana, Madrid.

Ángulo U., J. M., Romero Y., S. & Ángulo M., I. (2006). *Microcontroladores <<PIC>>. Diseño práctico de Aplicaciones. Segunda Parte: PIC16F87x*. Segunda Edición. McGraw-Hill Interamericana, Madrid.

Bates, M. (2013). *Interfacing PIC Microcontrollers: Embedded Design by Interactive Simulation*. Elsevier, Oxford, UK.

Hellebuyck, C. (2003). *Programming PIC Microcontrollers with PicBasic*. Elsevier Science, United States of America.

Maini, A. K. (2007). *Digital Electronics: Principles, Devices and Applications*. John Wiley & Sons, Ltd. Chichester, Inglaterra.

Pazmiño O., L. A. & Sánchez G., R. A. (2010). *Diseño y Construcción de un Prototipo Inalámbrico RF para el Monitoreo de la Seguridad Residencial de forma Remota empleando la Internet*. Proyecto de Grado de Ingeniería Electrónica y Telecomunicaciones. Escuela Politécnica Nacional (EPN), Quito.

Reyes, C. A. (2008). *Microcontroladores PIC: Programación en Basic*. Tercera Edición. RISPGRAF. Quito, Ecuador.

Sánchez, J. & Cantón, M. P. (2007). *Microcontroller Programming. The Microchip PIC*. CRC Press. United States of America.

Stallings, W. (2004). *Comunicaciones y Redes de Computadores*. Pearson Educación, Madrid.

Tanenbaum, A. S. (2003). *Redes de computadoras*. Pearson Educación, México.

Valdivia M., C. (2014). *Sistemas Informáticos y Redes Locales*. Ediciones Paraninfo S.A. Madrid, España.

Verle, M. (2010). *PIC Microcontrollers: Programming in Basic*. MikroElektronika, Belgrado.