



**UNIVERSIDAD MICHOACANA DE SAN NICOLÁS
DE HIDALGO**

FACULTAD DE INGENIERÍA ELÉCTRICA

**“IMPLEMENTACIÓN DE ESQUEMAS DE COMUNICACIÓN
PARA EL MONITOREO Y CONTROL REMOTO DE UN
SISTEMA DE NIVEL DE LÍQUIDO”**

TESIS

**Que para obtener el Título de
INGENIERO EN ELECTRÓNICA**

Presenta

ALEJANDRO BARAJAS CASTILLO

Asesor de Tesis

M.I. ISIDRO IGNACIO LÁZARO CASTILLO

Coasesor de Tesis

M.I. SALVADOR RAMÍREZ ZAVALA

Octubre del 2011

Agradecimientos

Quisiera expresar mi más profundo agradecimiento:

A mi madre Irma Marisol Castillo Mendoza y a mis abuelos María Guadalupe Mendoza Luviano y Roberto Castillo Ferreira por guiarme.

A la Universidad Michoacana por darme la gran oportunidad de estudiar una carrera.

Al maestro Salvador Ramírez Zavala por su apoyo incondicional, por dejarme aprender tanto de él, por compartir sus conocimientos y siempre guiarme.

Al Maestro Isidro Ignacio Lázaro Castillo y al Doctor Juan Anzurez Marín por su apoyo durante el desarrollo del trabajo.

A mis compañeros por aprender tanto de ellos y de nuestras inquietudes.

A mi compañera por apoyarme.

Dedicatoria

A mi madre Irma Marisol por su gran ejemplo de dedicación y estudio, por todo su apoyo incondicional, darme todo a su alcance y ayudarme a conseguir todas mis metas.

A mis abuelos María Guadalupe Mendoza Luviano y Roberto Castillo Ferreira por su gran ejemplo de trabajo, apoyo incondicional y dejarme aprender de ellos, les agradezco todo lo que hicieron.

A mis hermanos Roberto y César por siempre estar cuando más los necesite y aprender tanto de ellos.

Resumen

El presente trabajo muestra la instrumentación, el control y el monitoreo de las variables de un sistema de nivel de líquido de primer orden mediante el uso de 3 de los esquemas de comunicación más utilizados; Radiofrecuencia, GSM (Sistema global para las comunicaciones móviles) e Internet.

El sistema de nivel de líquido que se utilizó está compuesto por dos tanques interconectados (para fines de control solo se usa uno de ellos), el control de flujo del sistema se implementó mediante 2 electroválvulas, el monitoreo del nivel de líquido se realizó mediante 2 sensores de ultrasonido y el caudal de entrada del sistema mediante una bomba de inducción monofásica.

La aplicación de los esquemas se realizó mediante el diseño de 3 módulos que permiten el uso de un microcontrolador PIC18F4550 y un PIC18F2550, para controlar el sistema.

- Módulo ZigBee de Maxstream. Este tipo de dispositivos permiten una comunicación simple y confiable entre microcontroladores, computadoras y cualquier dispositivo vía inalámbrica. Se diseñaron 3 módulos inalámbricos con tecnología ZigBee, los cuales nos permiten monitorear y controlar el sistema de manera remota usando radiofrecuencia.
- Librerías TCP/IP y TELNET de LabVIEW®. Estas librerías nos permiten la transmisión de datos vía Internet. La recepción de datos se procesa desde un módulo diseñado en base al microcontrolador PIC18F2550.
- Módulo GSM/GPRS SM5100B. Este módulo contiene las aplicaciones de casi cualquier celular como SMS, GSM y TCP/IP, el cual nos permite el uso del esquema GSM y SMS.

Los módulos diseñados nos permitieron la transmisión y recepción de los datos que se necesitan para el monitoreo y el control remoto del sistema, los cuales se realizaron utilizando el lenguaje de programación gráfico LabVIEW®. El sistema se controla en base a un controlador PID implementado en LabVIEW®, las ganancias del controlador son calculadas en base a un modelo tomado de un trabajo anterior [Barajas, 2010].

Índice general

1. Introducción	1
1.1. Antecedentes	2
1.1.1. Historia de las comunicaciones	2
1.1.1.1. Comunicaciones electrónicas	2
1.1.1.2. Telégrafo	2
1.1.1.3. Teléfono	3
1.1.1.4. Comunicaciones inalámbricas	3
1.1.1.5. Radio	3
1.1.1.6. Digitalización	3
1.1.2. Conceptos básicos de comunicaciones	4
1.1.2.1. Comunicaciones electrónicas	4
1.1.2.2. Información	5
1.1.2.3. Mensaje	5
1.1.2.4. Sistemas de comunicaciones analógicos y digitales	6
1.1.2.5. Modulación y demodulación	6
1.1.2.6. Espectro electromagnético	7
1.1.2.7. Frecuencias de transmisión	8
1.1.2.8. Clasificación de transmisores	10
1.1.2.9. Ancho de banda	10
1.1.2.10. Capacidad de información	10
1.1.2.11. Modos de transmisión	10
1.2. Objetivo general	11
1.2.1. Objetivos particulares	11
1.3. Justificación	11
1.4. Metodología	12

1.5. Descripción de capítulos	13
2. Descripción de los Esquemas de Comunicación	15
2.1. Red de comunicación	16
2.1.1. Tareas de un sistema de comunicaciones	17
2.2. Bases teóricas del protocolo TCP/IP	20
2.2.1. Modelo OSI	20
2.2.2. Protocolo IP	22
2.2.3. Direcciones IP	22
2.2.4. Protocolo TCP	23
2.2.5. Protocolo TELNET	24
2.3. Bases teóricas del Sistema Global para las Comunicaciones Móviles	25
2.3.1. Antecedentes del estándar GSM	25
2.3.2. Estándar GSM	26
2.3.3. Comunicación a través de la red GSM.	27
2.3.4. Servicios del GSM	29
2.3.5. Servicio de Mensajes Cortos (SMS)	30
2.3.5.1. Elementos y arquitectura necesaria para el servicio SMS	31
2.3.5.2. Elementos de señalización	32
2.4. Introducción al protocolo ZigBee	33
2.4.1. Modos de operación	34
2.4.1.1. Modo RECIBIR/TRANSMITIR	34
2.4.1.2. Modo de bajo consumo (SleepMode)	35
2.4.1.3. Modo de comando	35
2.4.1.4. Modo transparente	35
2.4.1.5. Modo de operación API	35
2.4.1.6. Modo de operación IDLE	36
2.4.2. Direccionamiento de los módulos	36
2.4.3. Modos de conexión	36
2.4.3.1. Modo de conexión transparente	36
2.4.3.1.1. Modo de conexión punto a punto	36
2.4.3.1.2. Modo de conexión punto a multipunto	37
2.4.3.2. Modo de conexión broadcast	39

2.4.3.3.	Modo de conexión cable virtual I/O	39
2.4.4.	Modo de conexión NonBeacon. Peer toPeer	40
2.4.5.	Conexión NonBeacon con coordinador	41
2.4.6.	Conexión API	41
3.	Descripción del Sistema de Nivel de Líquido	43
3.1.	Descripción del sistema de nivel de líquido	44
3.1.1.	Sensores de ultrasonido Banner Q45U	45
3.1.1.1.	Electroválvulas Honeywell, modelo ML6984	46
3.1.2.	Bomba de agua	47
3.2.	Instrumentación del sistema de nivel de líquido	48
3.2.1.	Instrumentación de los sensores de ultrasonido Banner Q45U	48
3.2.1.1.	Etapas de alimentación del sensor de ultrasonido	48
3.2.1.2.	Acondicionamiento de la señal del sensor de ultrasonido	49
3.2.2.	Instrumentación de las Electroválvulas Honeywell, modelo ML6984	51
3.2.2.1.	Diseño del filtro pasa bajas	52
3.2.3.	Etapas de potencia para el arranque de la bomba de inducción	56
3.3.	Modelado del sistema	58
3.3.1.	Método de Ziegler & Nichols	58
3.3.2.	Modelado del sistema en lazo abierto	59
3.3.3.	Modelado del tanque 1.	61
3.3.4.	Modelado del tanque 2.	62
4.	Implementación de los Esquemas de Comunicación	63
4.1.	Requerimientos del sistema de nivel de líquido	64
4.1.1.	PIC18F4550	65
4.1.2.	PIC18F2550	66
4.2.	Tarjeta basada en el PIC18F2550	67
4.3.	Implementación del protocolo ZigBee	70
4.3.1.	Chip de radiofrecuencia Xbee®	71
4.3.1.1.	Comandos AT	72
4.3.1.2.	Modo de operación punto a punto	76
4.3.1.3.	Direcciones cortas de 16 bits	76

4.3.1.4.	Direcciones largas de 64 bits	77
4.3.1.5.	Configuración de los módulos Xbee®	77
4.3.2.	Tarjeta adaptadora del módulo Xbee®	78
4.3.3.	Tarjeta inalámbrica basada en el PIC18F4550	79
4.3.4.	Tarjeta inalámbrica basada en el PIC18F2550	85
4.3.5.	Tarjeta inalámbrica basada en el puerto RS232	89
4.3.6.	Funciones programadas	93
4.3.7.	Flujo básico de las rutinas para comunicación mediante el chip de radiofrecuencia Xbee®	97
4.4.	Implementación del esquema de comunicaciones TCP/IP	98
4.4.1.	Estación servidor TCP/IP	98
4.4.2.	Estación cliente TCP/IP	99
4.4.3.	Interconexión entre la computadora cliente y servidora vía TCP/IP	100
4.4.4.	Librerías encargadas de transmitir, recibir datos mediante TCP/IP	101
4.4.4.1.	Transmisión de datos vía TCP/IP	102
4.4.4.2.	Recepción de datos vía TCP/IP	103
4.4.5.	Librería encargada cerrar la conexión TCP/IP	104
4.5.	Implementación del esquema de comunicaciones TELNET	104
4.5.1.	Estación servidor TELNET	105
4.5.2.	Estación cliente TELNET	106
4.5.3.	Interconexión entre la computadora cliente y servidora vía TELNET	107
4.5.4.	Librerías encargadas de transmitir, recibir datos mediante TELNET	107
4.5.4.1.	Transmisión de datos vía TELNET	109
4.5.4.2.	Recepción de datos vía TCP/IP	110
4.5.5.	Librería encargada cerrar la conexión TELNET	110
4.6.	Flujo básico de las rutinas para comunicación TCP/IP y TELNET en Lab- VIEW®	111
4.6.1.	Establecimiento de la comunicación	111
4.6.2.	Envío de datos o escritura	112
4.6.3.	Diagrama de conexión del sistema mediante el esquema TCP/IP	113
4.6.4.	Diagrama de conexión del sistema mediante el esquema TELNET	114
4.7.	Implementación del esquema de comunicaciones GSM	115

4.7.1.	Módulo SM5100B	115
4.7.1.1.	Modo de comandos AT	120
4.7.2.	Tarjeta inalámbrica para el control del módulo GSM/GPRS SM5100B basada en el microcontrolador PIC18F2550	121
4.7.3.	Funciones programadas	125
4.7.3.1.	Funciones para configurar el módulo GSM	125
4.7.3.2.	Función para transmitir llamadas	128
4.7.3.3.	Función para transmitir mensajes SMS	130
4.7.3.3.1.	Modo de texto SMS	130
4.7.3.3.2.	Modo PDU SMS	131
4.7.3.3.3.	Configuración del módulo GSM para la transmi- sión de mensajes de texto	131
4.7.3.4.	Recepción de llamadas	132
4.7.3.5.	Recepción de mensajes SMS	133
4.7.4.	Flujo básico de las rutinas para comunicación mediante comunica- ción GSM	133
4.7.5.	Recepción de mensajes de texto	134
4.8.	Implementación del controlador PID	136
5.	Diseño de las Interfaces Gráficas	139
5.1.	Bytes a transmitir y recibir necesarios para comunicarse con el sistema . .	139
5.2.	Interfaz mediante el protocolo ZigBee	140
5.2.1.	Modo configuración de la interfaz mediante el esquema de radiofre- cuencia	141
5.2.1.1.	Diagrama de bloques del modo configuración de la interfaz mediante el esquema de radiofrecuencia	143
5.2.2.	Modo manual de la interfaz mediante el esquema de radiofrecuencia	143
5.2.2.1.	Diagrama de bloques del modo manual de la interfaz me- diante el esquema de radiofrecuencia	144
5.2.3.	Modo automático de la interfaz mediante el esquema de radiofre- cuencia	145
5.2.4.	Modo automático para el tanque 1 de la interfaz mediante el esque- ma de radiofrecuencia	146

5.2.5.	Modo automático para el tanque 2 de la interfaz mediante el esquema de radiofrecuencia	147
5.2.6.	Modo automático para el sistema de la interfaz mediante el esquema de radiofrecuencia	148
5.3.	Interfaz mediante los esquemas TCP/IP y TELNET	149
5.3.1.	Modo configuración de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora servidor	151
5.3.2.	Modo manual de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora servidor	152
5.3.3.	Modo automático para el tanque 1 de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora servidor	153
5.3.4.	Modo automático para el tanque 2 de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora servidor	154
5.3.5.	Modo automático para el monitoreo del sistema de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora servidor	156
5.3.6.	Modo configuración de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora cliente	157
5.3.7.	Modo automático para el tanque 1 de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora cliente	158
5.3.8.	Modo automático para el tanque 2 de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora cliente	160
6.	Pruebas y Resultados	162
6.1.	Pruebas de la conversión de la señal PWM a una señal analógica	162
6.2.	Monitoreo y control remoto mediante el protocolo ZigBee®	164
6.2.1.	Pruebas de llenado y vaciado del tanque 1	164
6.2.2.	Prueba en lazo abierto del tanque 1 a una entrada escalón de 8 volts sobre la electroválvula 1	165
6.2.3.	Prueba en lazo abierto del tanque 1 a una entrada escalón de 5 volts sobre la electroválvula 1	166
6.2.4.	Pruebas del controlador PID	166

6.2.4.1.	Prueba PID con una perturbación externa ganancias $K_p = 1.15, T_i = 20\text{seg}$ y $T_d = 5\text{seg}$.	167
6.2.4.2.	Prueba PID con un cambio de referencia ganancias $K_p = 1.15, T_i = 20\text{seg}$ y $T_d = 5\text{seg}$.	167
6.2.4.3.	Prueba PID con ganancias $K_p = 5, T_i = .333\text{min}$ y $T_d = .083\text{min}$.	168
6.2.4.4.	Prueba PID con ganancias $K_p = 8, T_i = .333\text{min}$ y $T_d = .083\text{min}$.	169
6.2.4.5.	Prueba PID con ganancias $K_p = 8, T_i = .3333\text{min}$ y $T_d = .9\text{min}$.	169
6.2.4.6.	Comparación de las pruebas realizadas	170
6.2.4.7.	Pruebas de distancia realizadas sobre los módulos Xbee-PRO®	170
6.2.4.8.	Pruebas de distancia realizadas sobre los módulos Xbee®	171
6.3.	Pruebas mediante el esquema TCP/IP	171
6.3.1.	Prueba PID con ganancias $K_p = 9, T_i = .4\text{min}$ y $T_d = .4\text{min}$ mediante el esquema TCP/IP	172
6.4.	Monitoreo remoto mediante el esquema TELNET	173
6.4.1.	Prueba PID con ganancias $K_p = 8, T_i = .333\text{min}$ y $T_d = .9\text{min}$ mediante el esquema TELNET	173
6.4.2.	Prueba PID con ganancias $K_p = 8, T_i = .333\text{min}$ y $T_d = .5\text{min}$ mediante el esquema TELNET	174
6.5.	Monitoreo remoto mediante el esquema GSM	175
6.5.1.	Transmisión de llamadas	175
6.5.2.	Transmisión de mensajes SMS	175
7.	Conclusiones	178
7.1.	Trabajos futuros	180

Apéndice A

LISTA DE SÍMBOLOS Y ABREVIATURAS	181
----------------------------------	-----

Apéndice B

COPIA DEL ARTÍCULO PUBLICADO. S. RAMIREZ, I. LÁZARO, A. BARAJAS, “INSTRUMENTACIÓN, MONITOREO Y CONTROL DE UN SISTEMA DE NIVEL DE LÍQUIDO VÍA INALÁMBRICA MEDIANTE LA TECNOLOGÍA ZIGBEE.” I CONGRESO INTERNACIONAL DE INGENIERÍA ELECTRÓNICA, INSTRUMENTACIÓN Y COMPUTACIÓN. MINATITLÁN VERACRUZ, MÉXICO	188
---	-----

Apéndice C

CARACTERES ASCII	196
------------------	-----

Apéndice D

CARACTERÍSTICAS DE LA ELECTROVÁLVULA, SENSOR DE ULTRASONIDO, MICROCONTROLADORES, MÓDULO GSM, MÓDULO XBEE® Y ESQUEMÁTICOS DE LAS TARJETAS REALIZADAS	204
---	-----

Apéndice E

PROGRAMAS PARA EL MANEJO DE LOS ESQUEMAS DE COMUNICACIÓN MEDIANTE LOS MICROCONTROLADORES	243
--	-----

Índice de figuras

1.1. Esquema de un sistema de comunicaciones unidireccional.	5
1.2. Esquema de un sistema de comunicaciones bidireccional.	5
1.3. Esquema simplificado de un sistema de comunicaciones.	7
1.4. Espectro electromagnético.	8
2.1. Modelo de un sistema de comunicaciones con una conexión punto a punto.	16
2.2. Diagrama de bloques simplificado para una comunicación a través de la red GSM.	26
2.3. Elementos de red GSM necesarios para proveer el servicio SMS.	32
2.4. Configuración para la operación punto a punto en los módulos Xbee®. . .	37
2.5. Configuración para la operación punto a multipunto en los módulos Xbee®.	37
2.6. Ancho de banda de los canales de frecuencia existentes en los módulos Xbee®.	38
2.7. Configuración Broadcast en los módulos Xbee®.	39
2.8. Vista superior del diagrama de pines del módulo Xbee®.	40
2.9. Estructura del frame del modo API en los módulos Xbee®.	41
3.1. Sistema de nivel de líquido.	44
3.2. Tanques para almacenamiento de fluidos con capacidad de 74 litros.	45
3.3. Depósito de agua con una capacidad de 100 litros.	45
3.4. Aspecto físico de los sensores ultrasónicos Banner Q45U.	45
3.5. Rango de operación de los sensores ultrasónicos Banner Q45U.	46
3.6. Aspecto físico de la electroválvula con su respectivo actuador.	47
3.7. Aspecto físico del motor de inducción.	47
3.8. Prueba de llenado del tanque 1.	47
3.9. Esquemático de la fuente de alimentación de los sensores de ultrasonido. .	48

3.10. Aspecto real de la fuente de alimentación de los sensores de ultrasonido. . .	49
3.11. Amplificador operacional inversor.	49
3.12. Arreglo de operacionales para reducir el voltaje de los sensores de ultrasonido.	50
3.13. Aspecto real del arreglo de amplificadores operacionales inversores utilizados.	51
3.14. Esquema seguido para el filtro pasa bajas	53
3.15. Esquema del filtro butterworth pasa bajas de segundo orden implementado.	55
3.16. Diagrama del amplificador no inversor utilizado.	55
3.17. Esquema del filtro pasa bajas de segundo orden a 33Hz implementado para cada una de las señales PWM generadas.	56
3.18. Aspecto real del filtro pasa bajas construido.	56
3.19. Esquemático del controlador del motor de inducción.	57
3.20. Aspecto real del controlador del motor de inducción.	57
3.21. Curva S, respuesta de un proceso típico a la entrada escalón unitario. . . .	58
3.22. Respuestas del sistema ante entradas escalón sobre la electroválvula 1. . . .	59
3.23. Método de Ziegler & Nichols aplicado a la prueba en lazo abierto 5V. . . .	60
3.24. Validación del modelo obtenido.	61
4.1. Esquema de conexión para el ajuste de las señales de los sensores de ultra- sonido.	64
4.2. Esquema de conexión para el ajuste de las señales de las electroválvulas. . .	65
4.3. Esquema de conexión seguido para el control de la bomba de agua.	65
4.4. Aspecto físico y diagrama de pines del PIC18F4550.	66
4.5. Aspecto físico y diagrama de pines del PIC18F2550.	67
4.6. Diagrama de bloques de la tarjeta basada en el PIC18F2550.	68
4.7. Aspecto real de la tarjeta diseñada basada en el PIC18F2550.	68
4.8. Esquema general de conexión del sistema mediante la tarjeta basada en el PIC18F2550.	70
4.9. Esquema general de conexión entre una computadora y un microcontrola- dor mediante módulos Xbee®.	71
4.10. Esquema de comunicación entre 2 microcontroladores.	71
4.11. Sintaxis para enviar comandos AT al módulo Xbee®.	72
4.12. Sintaxis para leer comandos AT al módulo.	73
4.13. Direccionamiento de 16bits de dos módulos Xbee®.	77

4.14. Configuración punto a punto utilizado para los módulos Xbee®.	78
4.15. Dimensiones del módulo XbeePRO®.	78
4.16. Dimensiones del módulo Xbee®.	78
4.17. Esquemático del adaptador para el módulo Xbee®.	79
4.18. Aspecto real del adaptador para el módulo Xbee®.	79
4.19. Diagrama de bloques de las tarjetas inalámbricas basadas en el módulo Xbee®.	80
4.20. Aspecto real de la tarjeta inalámbrica basada en el PIC18F4550.	80
4.21. Acoplo de la señal de 5 a 3.3 volts.	81
4.22. Aspecto real de la tarjeta inalámbrica basada en el PIC18F4550 con la tarjeta adaptadora del módulo Xbee®.	84
4.23. Esquema general de conexión del sistema mediante la tarjeta inalámbrica basada en el PIC18F4550.	85
4.24. Aspecto real de la tarjeta inalámbrica basada en el PIC18F2550.	85
4.25. Aspecto real de la tarjeta inalámbrica basada en el PIC18F2550 con la tarjeta adaptadora del módulo Xbee®.	88
4.26. Esquema general de conexión del sistema mediante la tarjeta inalámbrica basada en el PIC18F2550.	89
4.27. Diagrama de bloques de la tarjeta inalámbrica basada en el puerto RS232.	89
4.28. Aspecto real de la tarjeta inalámbrica basada la comunicación RS232.	90
4.29. Aspecto real de la tarjeta inalámbrica RS232 con la tarjeta adaptadora del módulo Xbee®.	92
4.30. Esquema general de conexión del sistema mediante la tarjeta inalámbrica RS232.	93
4.31. Bytes necesarios para entrar en línea de comandos AT del módulo Xbee®.	94
4.32. Bytes necesarios para entrar configurar la dirección interna del módulo Xbee®.	94
4.33. Bytes necesarios para modificar el canal de transmisión del módulo Xbee®.	95
4.34. Bytes necesarios para modificar el número de red del módulo Xbee®.	95
4.35. Bytes necesarios para modificar el número de red del módulo Xbee®.	96
4.36. Bytes necesarios para modificar el número de red del módulo Xbee®.	96
4.37. Bytes necesarios para modificar el número de red del módulo Xbee®.	97

4.38. Diagrama de flujo del esquema implementado basado en el módulo Xbee®.	97
4.39. Librería TCP/IP para esperar la conexión al puerto.	98
4.40. Librería TCP/IP para establecer la conexión con la computadora servidor.	99
4.41. Interconexión entre la computadora cliente y servidora vía TCP/IP.	100
4.42. Librería TCP/IP para transmitir datos.	101
4.43. Librería TCP/IP para recibir datos.	102
4.44. Transmisión de datos vía TCP/IP utilizando las librerías LabVIEW®.	103
4.45. Recepción de datos vía TCP/IP utilizando las librerías LabVIEW®.	103
4.46. Librería TCP/IP para cerrar la conexión.	104
4.47. Librería TELNET para esperar la conexión al puerto.	105
4.48. Librería TELNET para solicitar acceso a la computadora servidor.	106
4.49. Interconexión entre la computadora cliente y servidora vía TELNET.	107
4.50. Librería TELNET para transmitir datos.	107
4.51. Librería TELNET para recibir datos.	108
4.52. Transmisión de datos vía TELNET utilizando las librerías LabVIEW®.	109
4.53. Recepción de datos vía TELNET utilizando las librerías LabVIEW®.	110
4.54. Librería TELNET para cerrar la conexión.	110
4.55. Diagrama de flujo del establecimiento de conexión en la computadora ser- vidor.	112
4.56. Diagrama de flujo del establecimiento de conexión en la computadora cliente.	112
4.57. Diagrama de flujo de la transmisión de datos vía TCP/IP o TELNET.	113
4.58. Diagrama de flujo de la recepción de datos vía TCP/IP ó TELNET.	113
4.59. Monitoreo y control del sistema mediante la conexión TCP/IP.	114
4.60. Monitoreo y control del sistema mediante la conexión TCP/IP.	114
4.61. Aspecto real del módulo GSM SM5100B.	117
4.62. Conector de 60 pines para acceder a los pines del módulo SM5100B.	117
4.63. Aspecto real del conector SIM.	118
4.64. Aspecto real de la antena cuatri-banda para el módulo GSM.	118
4.65. Tarjeta utilizada para adquirir las señales del módulo GSM.	119
4.66. Esquema general de conexión entre una computadora y un microcontrola- dor mediante un módulo GSM SM5100B.	119

4.67. Esquema general de conexión entre un microcontrolador y el módulo GSM SM5100B.	119
4.68. Forma de introducción de comandos AT en el módulo GSM.	121
4.69. Diagrama de bloques de las tarjetas inalámbricas basadas en el módulo GSM/GPRS SM5100B.	121
4.70. Aspecto real de la tarjeta diseñada basada en el PIC18F2550 para el control del módulo GSM.	122
4.71. Esquema general de conexión del sistema mediante la tarjeta basada en el módulo GSM.	124
4.72. Trama de inicialización del módulo GSM.	125
4.73. Configuración de la fecha y hora en el módulo GSM.	127
4.74. Configuración de la frecuencia de trabajo del módulo GSM.	128
4.75. Transmisión de llamadas con el módulo GSM.	129
4.76. Configuración del módulo GSM para la transmisión de mensajes de texto. .	131
4.77. Transmisión de mensajes de texto.	132
4.78. Recepción de llamadas en el módulo GSM.	132
4.79. Recepción de mensajes SMS en el módulo GSM.	133
4.80. Diagrama de flujo del esquema implementado basado en esquema GSM. . .	133
4.81. Recepción de mensajes de texto en el módulo GSM.	134
4.82. Librería PID implementada para el control del sistema.	136
5.1. Esquema para el monitoreo y control con el módulo Xbee y el microcon- trolador PIC18F2550.	141
5.2. Esquema para el monitoreo y control con el módulo Xbee y el microcon- trolador PIC18F4550.	141
5.3. Aspecto de la interfaz gráfica diseñada para el esquema de radio frecuencia en su modo configuración.	142
5.4. Diagrama de bloques para el esquema de radio frecuencia en su modo con- figuración.	143
5.5. Aspecto de la segunda interfaz gráfica diseñada para el esquema de radio frecuencia en su modo manual.	144
5.6. Diagrama de bloques para el esquema de radio frecuencia en su modo manual.	145

5.7. Diagrama de bloques para el esquema de radio frecuencia en su modo automático.	145
5.8. Aspecto de la segunda interfaz gráfica diseñada para el esquema de radio frecuencia en su modo automático, para el tanque 1.	146
5.9. Aspecto de la segunda interfaz gráfica diseñada para el esquema de radio frecuencia en su modo automático para el tanque 2.	147
5.10. Aspecto de la segunda interfaz gráfica diseñada para el esquema de radio frecuencia en su modo automático, para el sistema completo.	149
5.11. Esquema para el monitoreo y control con el esquema TELNET.	150
5.12. Esquema para el monitoreo y control con el esquema TCP/IP.	150
5.13. Aspecto de la interfaz gráfica para la computadora servidor en su modo configuración mediante TCP/IP o TELNET.	151
5.14. Aspecto de la interfaz gráfica para la computadora servidor en su modo manual mediante TCP/IP o TELNET.	152
5.15. Aspecto de la interfaz gráfica para la computadora servidor en su modo automático para el tanque 1 mediante TCP/IP o TELNET.. . . .	153
5.16. Aspecto de la interfaz gráfica para la computadora servidor en su modo automático para el tanque 2 mediante TCP/IP o TELNET.	155
5.17. Aspecto de la interfaz gráfica para la computadora servidor en su modo automático para el sistema mediante TCP/IP o TELNET.	156
5.18. Aspecto de la interfaz gráfica para la computadora cliente en su modo manual mediante TCP/IP o TELNET.	157
5.19. Aspecto de la interfaz gráfica para la computadora cliente en su modo automático para el tanque 1 mediante TCP/IP o TELNET.	159
5.20. Aspecto de la interfaz gráfica para la computadora servidor en su modo automático para el tanque 2 mediante TCP/IP o TELNET.	160
6.1. Esquema de funcionamiento del filtro pasa bajas diseñado.	162
6.2. Prueba con una señal PWM a un ciclo de trabajo del 10 %.	163
6.3. Prueba con una señal PWM a un ciclo de trabajo del 50 %.	163
6.4. Prueba con una señal PWM a un ciclo de trabajo del 100 %.	164
6.5. Prueba de llenado del tanque 1.	164
6.6. Prueba de vaciado del tanque 1.	165

6.7. Prueba en lazo abierto ante una entrada escalón de 8 volts.	165
6.8. Prueba en lazo abierto ante una entrada escalón de 5 volts.	166
6.9. Prueba PID con módulo Xbee® $K_p = 1.15, T_i = .333min$ y $T_d = .0833min$	166
6.10. Prueba PID ante perturbación externa con módulo Xbee® $K_p = 1.15, T_i =$ $.333min$ y $T_d = .0833min$	167
6.11. Prueba PID ante cambio de referencia con módulo Xbee® $K_p = 1.15, T_i =$ $.333min$ y $T_d = .0833min$	168
6.12. Prueba PID con módulo Xbee® $K_p = 5, T_i = .333min$ y $T_d = .0833min$	168
6.13. Prueba PID con módulo Xbee® $K_p = 8, T_i = .333min$ y $T_d = .0833min$	169
6.14. Prueba PID con módulo Xbee® $K_p = 8, T_i = .333min$ y $T_d = .9min$	170
6.15. Comparación de las pruebas realizadas PID con módulo Xbee®.	170
6.16. Pruebas de distancia Xbee-PRO®.	171
6.17. Pruebas de distancia Xbee®.	171
6.18. Prueba PID mediante esquema TCP/IP® $K_p = 9, T_i = .4min$ y $T_d = .9min$	172
6.19. Prueba PID mediante esquema TCP/IP® $K_p = 9, T_i = .4min$ y $T_d = .9min$	172
6.20. Prueba PID mediante esquema TELNET $K_p = 8, T_i = .333min$ y $T_d = .9min$	173
6.21. Prueba PID mediante esquema TELNET $K_p = 8, T_i = .333min$ y $T_d = .9min$	174
6.22. Prueba PID mediante esquema TELNET $K_p = 8, T_i = .333min$ y $T_d = .5min$	174
6.23. Prueba de transmisión de llamadas mediante el módulo GSM.	175
6.24. Prueba 1 de transmisión de señales analógicas mediante mensajes de texto mediante el módulo GSM.	176
6.25. Prueba 2 de transmisión de señales analógicas mediante mensajes de texto mediante el módulo GSM.	176
6.26. Prueba 3 de transmisión de señales analógicas mediante mensajes de texto mediante el módulo GSM.	176
6.27. Prueba 1 de transmisión de la altura de los tanques mediante mensajes de texto mediante el módulo GSM.	177
6.28. Prueba 2 de transmisión de la altura de los tanques mediante mensajes de texto mediante el módulo GSM.	177

Índice de tablas

1.1. Rangos de frecuencia de las diferentes formas de ondas electromagnéticas. .	8
1.2. Nombres de las bandas de frecuencia del espectro electromagnético determinadas por el CCIR.	9
1.3. Frecuencias de los servicios de telecomunicaciones presentes en México. . .	9
2.1. Tareas básicas de un sistema de comunicaciones.	17
2.2. Capas del modelo OSI.	20
2.3. Modelo TCP/IP.	22
2.4. Estructura de las direcciones IP.	23
2.5. Canales de frecuencia a los que se puede tener acceso en los módulos Xbee®.	38
3.1. Tabla para el cálculo de las ganancias del controlador a utilizar.	59
4.1. Lista de comandos AT especiales para el módulo Xbee®.	73
4.2. Lista de comandos AT que permiten guardar las opciones seleccionadas para el módulo Xbee®.	74
4.3. Lista de comandos AT encargados de la interfaz serial para el módulo Xbee®.	74
4.4. Lista de comandos AT encargados de la red y de la seguridad de la red para el módulo Xbee®.	75
4.5. Ejemplo de configuración de los módulos Xbee® usando un direccionamiento de 16bits.	76
4.6. Formato de los comandos AT en su forma extendida para el módulo GSM 5100B-D.	120
4.7. Lista de comandos AT encargados de la configuración de la fecha y hora del módulo GSM.	126

4.8. Lista de comandos AT encargados de la configuración de la frecuencia del módulo GSM.	127
4.9. Lista de comandos AT encargados de la transmisión y finalización de una llamada telefónica en del módulo GSM.	128
4.10. Lista de comandos AT encargados de configurar, transmitir y recibir mensajes de texto SMS.	130
4.11. Estructura del mensaje seleccionado para controlar el sistema de nivel de líquido.	135
4.12. Estructura del mensaje para controlar la bomba de agua del sistema de nivel de líquido.	135
4.13. Estructura del mensaje para controlar la electroválvula 1 del sistema de nivel de líquido.	135
4.14. Estructura del mensaje para controlar la electroválvula 2 del sistema de nivel de líquido.	136
5.1. Estructura de la cadena de bytes transmitida por el microcontrolador. . . .	139
5.2. Estructura de la cadena de bytes recibida por el microcontrolador.	140

Capítulo 1

Introducción

Las comunicaciones inalámbricas son aquellas que no utilizan un medio de propagación físico, sino que utilizan ondas electromagnéticas con una cierta modulación, que se propagan por el aire, sin necesidad de una conexión directa con los dispositivos a comunicar [Tomasi, 2003].

En el mundo, las telecomunicaciones desde los años noventa son una parte muy importante de la vida cotidiana del hombre. Ya que están acaparando el interés del público en general. El mejor ejemplo lo constituyen los teléfonos móviles que han experimentado un crecimiento exponencial en la última década y se espera que, en un futuro cercano, lleguen a existir más de un billón de teléfonos móviles en todo el mundo.

Actualmente contamos con muchos protocolos de comunicación comerciales con los cuales muchas veces aún sin darnos cuenta, los utilizamos, ya que nos ayudan a hacer distintos tipos de tareas como lo son el Internet, los módulos GSM y la tecnología basada en radiofrecuencia. Estos protocolos nos permiten comunicarnos mandando mensajes de texto, transmisión de datos, servicios de voz y tener acceso a Internet.

Las comunicaciones inalámbricas componen uno de los sectores de la industria que mas rápido están creciendo. La velocidad y la distancia de cobertura de la comunicación inalámbrica, depende de forma exclusiva de la tecnología inalámbrica utilizada, así como de los dispositivos que forman parte de la red de comunicación [Aie, 2006].

El éxito de los esquemas de comunicación inalámbricos existentes son sus protocolos que son, determinadas reglas a cumplir por los dispositivos que desean comunicarse, como un idioma, los dispositivos deben aprender la gramática, la sintaxis y todas las reglas del idioma para poder comunicarse con otro dispositivo que habla ese idioma [Alipso, 2000].

1.1. Antecedentes

En este primer capítulo se darán algunos antecedentes históricos, conceptos básicos sobre comunicaciones y una breve explicación del contenido de esta tesis para facilitar el entendimiento de los capítulos siguientes. El propósito de esta sección es presentar una breve reseña histórica sobre las comunicaciones electrónicas.

1.1.1. Historia de las comunicaciones

Desde la antigüedad existieron sistemas de comunicación a distancia, en su mayoría haciendo uso del fuego mediante el humo. Durante largo tiempo el correo fue la única forma de comunicación a gran distancia, incorporando las mejoras que iban ofreciendo los nuevos adelantos tecnológicos: del caballo se pasó a los barcos y los ferrocarriles, después a los automóviles y por último a los aviones.

La historia de las comunicaciones electrónicas comenzó a mediados del siglo XIX [Tomasi, 2003].

1.1.1.1. Comunicaciones electrónicas

1865: El primer avance fue de tipo teórico, ya que las investigaciones matemáticas de James Clerk Maxwell indicaron que la energía electromagnética puede viajar por el espacio libre en forma de ondas o radiación en forma similar a como lo hace la luz.

1888: El científico alemán Heinrich Hertz comprueba en forma experimental las ecuaciones de Maxwell.

1892: E. Branly desarrolló el primer detector de radiación electromagnética basado en un “coherer”.

1.1.1.2. Telégrafo

1837: Samuel Morse desarrolló el primer sistema de comunicaciones utilizando inducción electromagnética.

1.1.1.3. Teléfono

1855: Antonio Meucci une dos habitaciones mediante una línea de voz usando guías de ondas acústicas.

1876: Alexander Graham Bell y su asistente Thomas Watson transmitieron una conversación humana usando cables metálicos como medio de transmisión.

1.1.1.4. Comunicaciones inalámbricas

1894: Guglielmo Marconi logró la primera transmisión telegráfica inalámbrica, usando señales de radio a una distancia de tres cuartos de Milla.

1899: Marconi envía el primer mensaje a través del Canal de la Mancha de Francia a Dover, Inglaterra.

1901: Marconi hace posible las primeras señales atravesando el Océano Atlántico.

1908: Lee de Forest inventa el tríodo de tubo de vacío (bulbo) basado en el efecto termiónico descubierto por Edison, el cual permite la primera amplificación práctica de señales electrónicas así como la modulación. Así nace la Electrónica. El tríodo hace posible la modulación de amplitud (AM) y con ello las transmisiones regulares de radio.

1948: William Bradford Shockley inventa el transistor, el cual hace el mismo trabajo que el tríodo, pero con un consumo de potencia miles de veces menor.

1.1.1.5. Radio

1920: Comienzan las primeras transmisiones de radio en AM comercial en Detroit y Pittsburgh.

1933: Edwin H. Armstrong inventa la Frecuencia Modulada (FM), comenzando la emisión comercial en FM en 1936.

1.1.1.6. Digitalización

1969: En Estados Unidos el departamento de defensa creó una red experimental de conmutación de paquetes utilizando las líneas telefónicas. Años más tarde esta red sería

el primer paso para Internet. Actualmente Internet se dice que tiene más de 1100 millones de usuarios.

1980: Aparece la Fibra Óptica que permiten la transmisión de luz con pérdidas mínimas a largas distancias con potencias pequeñas.

1982: El estándar GSM fue desarrollado. Actualmente este estándar cuenta con más de 3000 millones de usuarios en 212 países es el estándar más usado a nivel mundial para enviar y recibir mensajes por e-mail, faxes, navegar por Internet, así como utilizar otras funciones digitales de transmisión de datos mediante mensajes cortos SMS.

2003: El estándar IEEE 802.15.4 es aprobado.

2004: ZigBee es puesto a disposición del público.

1.1.2. Conceptos básicos de comunicaciones

Los conceptos de los diferentes esquemas de comunicaciones existentes no han cambiado mucho desde sus inicios. Son los métodos y la tecnología a través de la cual estos conceptos se implementan los que han sufrido cambios dramáticos y sorprendentes generando los avances actuales que van desde las comunicaciones por radiofrecuencia, la telefonía celular, el Internet, el GPS, etc. No hay límites sobre las expectativas para los sistemas de comunicaciones electrónicas del futuro [Tomasi, 2003].

1.1.2.1. Comunicaciones electrónicas

En esencia las comunicaciones electrónicas son la transmisión, recepción y procesamiento de información mediante circuitos electrónicos. Un sistema de comunicaciones electrónicas está compuesto por la fuente de información, el transmisor, el medio de transmisión, el receptor y el destino de la información, aunque frecuentemente estos elementos son resumidos únicamente en 3 de ellos, el transmisor, un medio de transmisión y el receptor. En la Figura 1.1 y Figura 1.2 se muestran los diagramas de bloques simplificados de los sistemas de comunicaciones electrónicas que existen mostrando la relación entre la información de la fuente original, el transmisor, el medio de transmisión, el receptor y la información recibida en el destino. El sistema de la Figura 1.1 es capaz de transmitir

información solamente en una dirección también (de la estación A a la estación B) este tipo de sistemas son también conocidos como sistema de comunicaciones unidireccionales, mientras que el sistema de comunicaciones mostrado en la Figura 1.2 es capaz de transmitir información en ambas direcciones (de la estación A a la B y de la estación B a la A) este tipo de sistemas son también conocidos como sistema de comunicaciones bidireccionales [Tomasi, 2003]. En todo sistema el transmisor es el encargado de convertir la

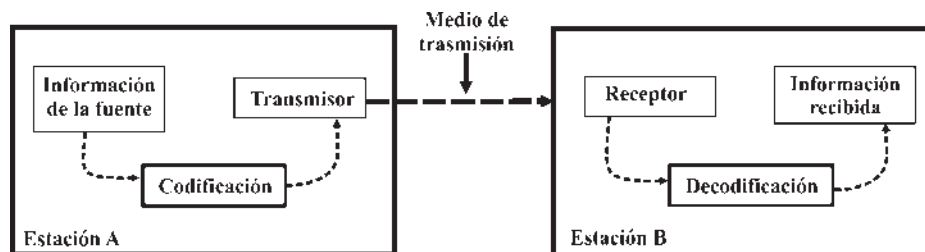


Figura 1.1: Esquema de un sistema de comunicaciones unidireccional.

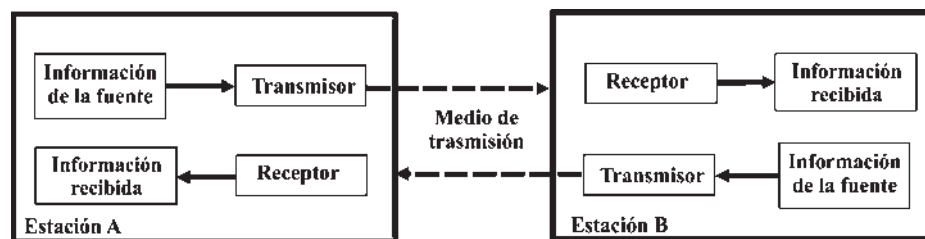


Figura 1.2: Esquema de un sistema de comunicaciones bidireccional.

información original en una forma más adecuada para su transmisión, el medio de transmisión proporciona un medio de conexión entre el transmisor y el receptor. El receptor es el encargado de convertir la información recibida a su forma original y transferirla a su destino.

1.1.2.2. Información

La información se define como el conjunto organizado de datos que contienen algún conocimiento (mensaje) o representación de algún aspecto de la realidad o la imaginación en la forma de: sonido, imagen, símbolos, códigos [Tomasi, 2003].

1.1.2.3. Mensaje

Como se muestra en la Figura 1.1, todo mensaje o información que se quiera transmitir debe ser convertida a energía electromagnética para poder ser transmitida mediante

dispositivos electrónicos. El proceso de convertir la información de su forma original a su forma transmisible es un tipo de codificación. A su vez en la recepción el mensaje debe ser reconvertido a su forma original (decodificación) [Tomasi, 2003].

1.1.2.4. Sistemas de comunicaciones analógicos y digitales

En los sistemas de comunicaciones analógicos la energía electromagnética se transmite y recibe en forma analógica (una señal variando continuamente tal como una onda senoidal).

Los sistemas de comunicación digitales son más complejos pero más confiables. En estos sistemas la energía electromagnética se transmite en forma de señal digital. (Una señal digital es una señal discontinua que normalmente toma sólo dos posibles valores). Las comunicaciones digitales abarcan un área extensa de técnicas de comunicaciones, algunas de ellas son la transmisión digital y la radio digital.

Los sistemas de comunicaciones analógicos fueron los primeros en desarrollarse, sin embargo los sistemas de comunicaciones digitales se han hecho más comunes. Con el tiempo los sistemas digitales han ido reemplazando a los analógicos, los cuales probablemente desaparezcan en un futuro [Tomasi, 2003].

1.1.2.5. Modulación y demodulación

Las señales de baja frecuencia (menos de 500 KHz) son difíciles de transmitir en forma de señales electromagnéticas ya que el rango de frecuencias de las señales a transmitir se traslapa y por lo tanto se interfieren: por ejemplo (señales de voz, de música, de mediciones, comparten el rango de 0 a 15 KHz) ya que las antenas requeridas son de tamaño proporcional a la longitud de onda (inversamente proporcional a la frecuencia).

La modulación permite “montar” la información (de baja frecuencia) a transmitir sobre una señal de alta frecuencia que es más fácil de radiar. A la señal a transmitir se le llama señal de modulación, modulante o señal de banda base y a la señal modulada de alta frecuencia se le llama portadora. Al proceso inverso de separar la señal de modulación de la portadora a partir de la señal modulada se le llama demodulación.

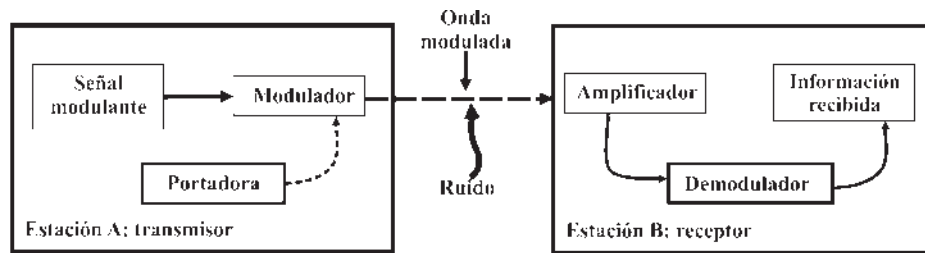


Figura 1.3: Esquema simplificado de un sistema de comunicaciones.

La Figura 1.3 es un diagrama a bloques simplificado de un sistema de comunicaciones que muestra la relación entre la señal de modulación (información). La señal modulada (portadora), la onda modulada (resultante) y el ruido que generalmente que introduce en el sistema durante el proceso [Tomasí, 2003].

1.1.2.6. Espectro electromagnético

La energía electromagnética puede viajar a la velocidad de la luz: como ondas de radio por el aire o por el vacío, como corriente eléctrica a través de un cable metálico, como luz a través de fibras ópticas, etc.

El sonido no es propiamente energía electromagnética sino más bien variaciones de presión, por lo tanto su velocidad de propagación depende enormemente del medio de transmisión.

La energía electromagnética se presenta en un rango de frecuencias casi infinito denominado espectro de frecuencias o espectro electromagnético. El espectro electromagnético se divide en dos grandes partes [Tomasí, 2003].

Ondas materiales: Se propagan por vibraciones de la materia (sólida, líquida o gaseosa). Incluyen:

- **Infrasonido** (debajo de los 8 Hz): Producido por algunos animales como los elefantes.
- **Sonido** (entre 8 y 30 KHz): Por ejemplo la voz humana (.3KHz a 4 KHz), música, señales telefónicas, etc.
- **Ultrasonido** (de los 30 KHz a los 20 MHz): Se usa en la inspección de la calidad de soldaduras, medición de distancia, aplicaciones médicas, etc.

Ondas electromagnéticas: Se generan debido a la variación de un campo electromagnético, y no requieren ningún tipo de soporte material para su propagación, por ejemplo:

- **Ondas de radio** (o Hertzianas): Microondas, Ondas luminosas (luz), Rayos X, Rayos gamma.

El espectro electromagnético se suele representar en una recta de frecuencia (Hz) como se muestra en la Figura 1.4.

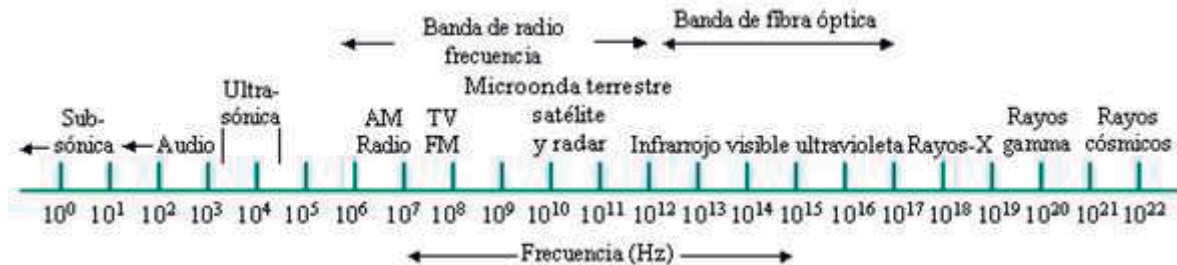


Figura 1.4: Espectro electromagnético.

Los rangos de frecuencia del espectro electromagnético se presentan en la Tabla 1.1.

Tabla 1.1: Rangos de frecuencia de las diferentes formas de ondas electromagnéticas.

Tipo de onda.	Frecuencia (Hertz)
Infrasonido.	0-8
Sonido.	$8 - 30 \times 10^3$
Ultrasonido.	$30 \times 10^3 - 20 \times 10^6$
Ondas de Radio (RF).	$0 - 1 \times 10^9$
Microondas.	$1 \times 10^9 - 3 \times 10^{12}$
Infrarrojo.	$3 \times 10^{12} - 4.6 \times 10^{14}$
Luz visible.	$4.6 \times 10^{14} - 7.5 \times 10^{14}$
Ultravioleta.	$7.5 \times 10^{14} - 6 \times 10^{16}$
Rayos x.	$6 \times 10^{16} - 1 \times 10^{20}$
Rayos Gamma	$1 \times 10^{20} - \dots$

1.1.2.7. Frecuencias de transmisión

El espectro electromagnético se divide en sectores o bandas de frecuencia. Cada banda tiene su nombre y sus límites.

El uso de las diferentes bandas para la transmisión es regulada por organismos de gobierno:

- La Comisión Federal de Telecomunicaciones (COFETEL) en México
- La Federal Communications Comision (FCC) en los EUA.

Existen también acuerdos internacionales: cada 10 años se realizan las Convenciones Internacionales de Telecomunicaciones.

El espectro de Radio Frecuencia se ha dividido en bandas designadas con nombres específicos por el CCIR (Comité Consultivo Internacional de Radio) como se muestra en la Tabla 1.2.

Tabla 1.2: Nombres de las bandas de frecuencia del espectro electromagnético determinadas por el CCIR.

Nombre	No. de Banda.	Abreviatura.	Frecuencias.
Extremely Low Frequency (Frecuencia Extremadamente Baja)	2	ELF	3-30 Hz
Voice Frequency (Frecuencia de Voz)	3	VF	0.3-3 KHz
Very Low Frequency (Frecuencia muy Baja)	4	VLF	3-30 KHz
Low Frequency (Frecuencia Baja)	5	LF	30-300 KHz
Medium Frequency (Frecuencia Media)	6	MF	300-3000 KHz
High Frequency (Frecuencia Alta)	7	HF	3-30 MHz
Very High Frequency (Frecuencia muy Alta)	8	VHF	30-300 MHz
Ultra High Frequency (Frecuencia Ultra Alta)	9	UHF	300-3000 MHz
Super High Frequency (Frecuencia Super Alta)	10	SHF	3-30 GHz
Extremely High Frequency (Frecuencia Extremadamente Alta)	11	EHF	30-300 GHz

Tabla 1.3: Frecuencias de los servicios de telecomunicaciones presentes en México.

Frecuencia	Servicio
800 MHz	Celulares.
900 MHz	Teléfonos inalámbricos de casa.
530 a 1700KHz	Banda de radio AM.
88 a 108MHz	Banda de radio FM.

La Tabla 1.3 muestra las frecuencias en las que trabajan algunos servicios de telecomunicaciones en México.

1.1.2.8. Clasificación de transmisores

Los transmisores se clasifican de acuerdo a lo siguiente:

- **Ancho de banda:** Rango de frecuencia en el que operan.
- **Tipo de modulación:** De amplitud, de frecuencia, de fase, de pulsos, no modulada.
- **Tipo de señal:** Analógica, digital o ambas.
- **Tipo de información que transmiten:** Telemetría, Telefonía, Radiodifusión de sonido, radiodifusión de vídeo.

1.1.2.9. Ancho de banda

El Ancho de Banda de un sistema de comunicaciones es el rango de frecuencias requerido para propagar la información de la fuente.

Este debe ser lo suficientemente ancho para pasar todas las frecuencias importantes de la información. Por ejemplo: para transmitir señales telefónicas de voz con buena calidad se requiere un ancho de banda mínimo de 3 KHz. Para transmitir señales de música de alta fidelidad por radio comercial FM se requiere de 200 KHz de ancho de banda. Para señales de TV se requiere un ancho de banda de casi 6 MHz.

1.1.2.10. Capacidad de información

La capacidad de información de un sistema de comunicaciones es una medida de la “cantidad de información” que el sistema puede transportar en un periodo dado de tiempo [Tomasi, 2003].

1.1.2.11. Modos de transmisión

Los sistemas de comunicaciones pueden diseñarse para transmitir información en una sola dirección, o en ambas direcciones [Tomasi, 2003]:

- **Simplex (SX):** Transmite en un solo sentido.
- **Half Duplex (HDX):** Transmite en ambas direcciones, pero no al mismo tiempo.
Ejemplo: radio de banda civil.

- **Full Duplex (FDX):** Transmite y recibe al mismo tiempo. Ejemplo: los sistemas telefónicos.
- **Full/Full Duplex (F/FDX):** Transmite y recibe al mismo tiempo, pero no necesariamente de la misma estación a la que transmite.

1.2. Objetivo general

Implementar 3 de los esquemas de comunicación más utilizados: radiofrecuencia, GSM (sistema global para las comunicaciones móviles) e Internet, mediante el desarrollo de tarjetas inalámbricas basadas en microcontroladores que permitirán hacer uso de cada uno de estos esquemas para el monitoreo de las variables y control de un sistema de nivel de líquido (de escala industrial) de primer orden.

1.2.1. Objetivos particulares

- Diseño e implementación de módulos basados en tecnología ZigBee, un microcontrolador PIC18F4550, un microcontrolador PIC18F4250 y comunicación RS232 para el monitoreo y control remoto del sistema de nivel de líquido.
- Implementación de las librerías TCP/IP y TELNET de LabVIEW® para el monitoreo y control del sistema de nivel de líquido desde Internet.
- Implementación del módulo GSM/GPRS SM5100B para el monitoreo y control manual del sistema de nivel de líquido de primer orden.
- Diseño de una interfaz gráfica basada en LabVIEW® amigable para el monitoreo y control del sistema por cada esquema implementado.

1.3. Justificación

En las comunicaciones los medios de transmisión son una parte esencial pero como todo componente de un sistema de computación, solos no sirven mucho.

Todos los esquemas de comunicación tienen una parte de software y hardware los cuales pueden llegar a ser usados y configurados mediante las diferentes herramientas y

dispositivos que se pueden encontrar en el mercado actual y que nos ofrecen una amplia gama de posibilidades a la hora de hacer alguna aplicación en específico. El conocimiento oportuno y completo de los esquemas de comunicación nos dan la ventaja de saber exactamente qué podemos hacer con lo que tenemos y que es lo mejor que podemos sacar de ellos.

El presente trabajo propone el control y monitoreo de las variables de un sistema de nivel de líquido de primer orden (de escala industrial) mediante el uso de 3 de los esquemas de comunicación más utilizados y populares: radiofrecuencia, GSM (Sistema global para las comunicaciones móviles) e Internet.

La elección de estos esquemas se debe a que en la actualidad casi cualquier persona tiene acceso a cualquiera de estos esquemas de comunicaciones ya sea desde una computadora, un celular, un radio por citar algunos dispositivos. También debido a que prácticamente en cualquier sistema industrial es fundamental el monitoreo y control remoto del sistema ya sea por seguridad, ya que estos sistemas pueden llegar a estar en lugares no accesibles y en otros casos, a la facilidad para manipular el sistema ya que no es necesario estar físicamente en él y a un interés personal para probar la versatilidad, gama de aplicaciones que se pueden llegar a tener, alcance, frecuencia a la que trabajan y el costo para llegar a manejar estos esquemas en aplicaciones específicas.

1.4. Metodología

El método empleado para el desarrollo de la tesis fue dividir los problemas básicos del control y monitoreo del sistema de mayor a menor, esto quiere decir que se tomaron parte por parte las necesidades y se clasificaron las tareas que se desarrollaron en prioridades y se siguieron. Estas tareas se clasifican en diez que son:

- Análisis de las diferentes tecnologías inalámbricas que se quieren implementar existentes (características, funcionalidad).
- Realización de un estudio de mercado de los componentes asociados a cada una de las tecnologías a implementar para poder determinar que módulos comprar y usar.
- Revisión bibliográfica y aprendizaje del lenguaje de programación a emplear para programar los módulos Xbee®[®], GSM SM5100B y librerías TCP/IP y TELNET de

LabVIEW®.

- Diseño, construcción e implementación de los módulos Xbee® para el monitoreo y control del sistema de nivel de líquidos.
- Implementación de las librerías TCP/IP y TELNET para el monitoreo y control del sistema de nivel de líquidos.
- Implementación del módulo GSM SM5100B para el monitoreo y control manual del sistema de nivel de líquidos.
- Desarrollo de la interfaz gráfica en LabVIEW® para el monitoreo y control del nivel de líquidos mediante los módulos Xbee®.
- Desarrollo de la interfaz gráfica para el monitoreo y control del nivel de líquidos mediante las librerías de TCP/IP y TELNET de LabVIEW®.
- Descripción del sistema de nivel de líquidos y sensores que lo componen.
- Pruebas y resultados.

1.5. Descripción de capítulos

Capítulo 1: se presenta, una breve reseña histórica y conceptos básicos sobre comunicaciones electrónica, así como la metodología, los objetivos y la justificación que tiene esta investigación.

Capítulo 2: se describen el funcionamiento general de los esquemas de comunicación que serán implementados los cuales son; GSM, radiofrecuencia e Internet.

Capítulo 3: se describe el sistema de nivel de líquidos de primer orden que se va controlar y monitorear así como los sensores y actuadores que lo componen.

Capítulo 4: se describe el diseño, uso y aplicación de los módulos Xbee®, de Internet y GSM SM5100B que son implementados para el monitoreo y control del sistema de nivel de líquido.

Capítulo 5: se presenta la descripción de las interfaces gráficas desarrolladas.

Capítulo 6: se presentan las pruebas realizadas sobre los esquemas así como los resultados obtenidos.

Capítulo 7: se muestran las conclusiones obtenidas, las desventajas de los esquemas empleados y las sugerencias para futuras investigaciones en este campo.

Capítulo 2

Descripción de los Esquemas de Comunicación

En junio de 1973 se organizó un seminario en la Universidad de Stanford para hacer frente al diseño de la propuesta de Internet y su protocolo. Un año después la versión inicial de TCP se especificó. Para mejorar la flexibilidad del protocolo de comunicación, en Enero de 1978 se propuso la división del protocolo TCP en dos componentes: un protocolo host-a-anfitrión dentro de las redes (TCP) y un protocolo de interconexión de redes (IP). El par de protocolos que se conoce como TCP/IP. IP permite transmitir paquetes individuales entre las máquinas, mientras que TCP se hace responsable de ordenar estos paquetes y proporcionar conexiones fiables entre los hosts.

Los desarrollos de GSM arrancan al principio de los 80's en un intento de resolver problemas surgidos por distintos sistemas de comunicaciones móviles nacionales. Los cuales implicaban la imposibilidad de utilizar un celular en diferentes países, así como la imposibilidad de crear un mercado común.

ZigBee es el nombre del conjunto de protocolos de comunicación inalámbrica para su utilización con radiodifusión digital de bajo consumo, basada en el estándar IEEE 802.15.4 de redes inalámbricas de área personal (Wireless Personal Area Network, WPAN). Su objetivo son las aplicaciones que requieren comunicaciones seguras con baja tasa de envío de datos y maximización de la vida útil de sus baterías.

En este capítulo se encuentra el marco teórico en el que se basa la tesis. Comienza con la explicación de los sistemas de comunicación y se prosigue con la explicación general de los esquemas de comunicación que se van a implementar.

2.1. Red de comunicación

Antes de realizar cualquier aplicación es necesario saber en donde se utiliza una red de comunicaciones. El propósito fundamental de una red de comunicaciones es el intercambio de datos entre dos o más sistemas. En la Figura 2.1 se muestra el modelo básico punto a punto enfocado en un ejemplo de intercambio de datos entre un dispositivo personal y un servidor a través de algún protocolo de comunicación.

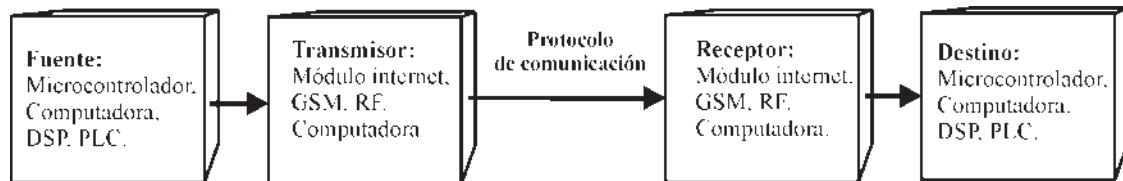


Figura 2.1: Modelo de un sistema de comunicaciones con una conexión punto a punto.

Los elementos de un modelo de comunicaciones son: la fuente, el transmisor, el receptor y el destino.

- Fuente: es la parte del sistema que genera los datos que serán transmitidos.
- Transmisor: es el encargado de transformar y codificar la información a través de algún protocolo de comunicación.
- Protocolo de comunicación: este puede llegar a ser desde una simple línea de transmisión de datos compuesta por dos cables de cobre hasta una red compleja que enlaza a la fuente con el destino.
- Receptor: se encarga de aceptar la señal de transmisor y la convierte en una señal que pueda ser procesada por el módulo destino.
- Destino: se encarga de recibir los datos provenientes del receptor.

Esta simple idea de un modelo punto a punto, en realidad contempla una complejidad técnica de acuerdo a la aplicación que se va a realizar y al módulo a utilizar para aplicar el protocolo de comunicaciones que se desea implementar. Para tener una idea de la complejidad del esquema de comunicaciones a implementar se presenta la Tabla 2.1 la cual contiene la lista de las tareas claves que deben de funcionar en un sistema de comunicación de datos [Bautista, 2005].

Tabla 2.1: Tareas básicas de un sistema de comunicaciones.

Tareas básicas de un sistema de comunicaciones.
Utilización del sistema de transmisión.
Interfase.
Generación de señales.
Sincronización.
Administración de intercambios.
Detección y corrección de errores.
Control de flujo.
Direccionamiento.
Ruteo.
Recuperación.
Formato de mensajes.
Seguridad
Administración de la red.

2.1.1. Tareas de un sistema de comunicaciones

El primer punto de la Tabla 2.1, la utilización del sistema de transmisión se refiere a la necesidad de hacer eficiente el uso de los sistemas de transmisión con los que se cuenta ya que normalmente se cuenta con más de un dispositivo para llevar a cabo el esquema de comunicación. Se debe de buscar la técnica más adecuada para realizar la transmisión de datos para que el sistema trabaje de manera adecuada y evitar que el módulo que se va a implementar no esté sobresaturado por una demanda excesiva de transmisión de datos [Bautista, 2005].

Para comunicarse un módulo o dispositivo debe hacer interfaz con el sistema de transmisión. La mayoría de las formas de comunicación dependen del uso de señales electromagnéticas que son propagadas sobre un medio de transmisión. Entonces, cuando la interfaz es establecida, la generación de señales es requerida para las comunicaciones. Las propiedades de las señales, tales como forma e intensidad, deben ser tales para que la señal sea capaz de ser propagada a través de un sistema de transmisión y además deben ser interpretadas como datos por el receptor (por lo general las señales tales como forma e intensidad de la señal a transmitir dependen y son generadas directamente por del modulo que se va a implementar).

No solamente las señales deben ser generadas para conformar los requerimientos del sistema de transmisión y para el receptor, también debe haber una forma de sincronización entre el transmisor y el receptor. El receptor debe ser capaz de determinar cuándo una

señal comienza y cuando termina. También debe conocer la duración de cada elemento de la señal.

Hay una variedad de requerimientos para una comunicación entre dispositivos, en la Tabla 2.1 éstos se concentran en la administración de intercambios. Si los datos deben ser intercambiados en ambas direcciones dentro de un periodo de tiempo, las dos partes deben cooperar. Para los dispositivos de procesamiento, será necesario más que una simple conexión establecida, se necesitan determinados requerimientos que deben ser definidos. Estos requerimientos pueden incluir entre otros, si ambos dispositivos pueden transmitir simultáneamente o deben tomar turnos, la cantidad de datos a ser enviados al mismo tiempo, el formato de los datos y qué hacer ante determinadas contingencias como la llegada de un error.

En todos los sistemas de comunicaciones hay un potencial de error; las señales transmitidas son distorsionadas antes de alcanzar su destino. La detección y corrección de errores son requeridas en circunstancias en donde los errores no pueden ser tolerados. Este es el típico caso de los sistemas de procesamiento de datos, donde los datos enviados no pueden ser alterados en el proceso y tienen que llegar como fueron enviados al destino.

El control de flujo es requerido para asegurar que la fuente no sobresature al destinatario al enviar los datos más rápido de lo que pueden ser procesados o recibidos.

El direccionamiento y ruteo se utilizan cuando más de 2 dispositivos comparten un sistema de transmisión, un sistema debe indicar la identidad del destinatario. El sistema de transmisión debe de asegurar que el sistema destinatario, y solo aquel sistema, reciba los datos. Además, el sistema de transmisión puede por sí mismo ser una red a través de la cual se puede tomar varios caminos. Una ruta específica de la red debe ser escogida. Es aquí donde se debe escoger la ruta óptima por la cual la información será enviada para ir desde el punto fuente hacia el punto destino.

La recuperación es un concepto diferente a la corrección de errores. Las técnicas de recuperación son necesarias en situaciones en donde un intercambio de información entre el transmisor y el receptor es interrumpido debido a una falla en alguna parte del sistema. El objetivo de la recuperación es que el sistema de comunicaciones sea capaz de resumir la actividad en el punto de interrupción o cuando menos restablecer el estado del sistema en la condición que se encontraba antes de que se presentara el error.

El formato de mensajes tiene que ver con el consentimiento de las dos partes (trans-

misor/receptor) en la definición de la manera en que serán intercambiados o transmitidos los datos. Es fundamental que en un sistema de comunicaciones el receptor entienda los datos de manera adecuada.

Frecuentemente es importante proveer algunas medidas de seguridad en los sistemas de comunicación de datos. El emisor de datos requiere estar seguro de que solo el receptor elegido sea quien recibe los datos. Mientras que el receptor de datos desea estar seguro de que los datos recibidos no hayan sido alterados durante la transmisión y que los mismos vienen del emisor adecuado.

Finalmente, dado que el intercambio de datos es un sistema complejo que no se puede ejecutar por sí mismo, se requiere de un administrador de la red. La administración de la red debe de ser capaz de configurar el sistema, monitorear su estado y reaccionar ante fallas que se puedan presentar [Bautista, 2005].

Como ya se mencionó, este trabajo plantea el desarrollo de un sistema de comunicaciones bidireccional basándonos en el sistema de la Figura 2.1 para el monitoreo y control remoto de un sistema de nivel de líquido mediante los esquemas de comunicación GSM, Internet y radiofrecuencia. Los módulos y librerías que se utilizan se encargan de cuidar las diferentes tareas de un sistema de comunicaciones ya mencionadas, como el control de flujo, la generación de señales, etc. Lo cual nos permite trabajar directamente con los datos que se quieran transmitir y con la aplicación física que en este caso es el sistema de nivel de líquido y las señales que lo componen. Las tarjetas diseñadas para el sistema tienen como elemento central 2 microcontroladores uno es el PIC18F4550 y el otro es el PIC18F2550, los cuales hacen uso del módulo GSM/GPRS SM5100B, las librerías TCP/IP, TELNET y del módulo en base a tecnología ZigBee para conectarse a los esquemas de comunicación deseados. Estos módulos siguen las tareas básicas de todo sistema de comunicaciones ya que implementan el protocolo necesario para conectarse a la red GSM, Internet y radiofrecuencia además de incluir protocolos como el RS232 que facilitan su interfaz con el microcontrolador, este hecho evita el tener que diseñar un sistema que permita conectarse a la red GSM o Internet desde cero, sin embargo, es necesario mencionar en forma breve los aspectos más importantes de los esquemas de comunicación GSM, Internet y ZigBee. Las secciones 2.2, 2.3 y 2.4 cumplen con tal propósito.

2.2. Bases teóricas del protocolo TCP/IP

TCP/IP corresponde a un conjunto de protocolos de comunicación para redes de computadoras, en el cual Internet está basado. Se denomina TCP/IP por dos de sus protocolos más importantes TCP (Transmission Control Protocol) e IP (Internet Protocol). Puede utilizarse prácticamente en todo tipo de redes de computadores, debido a su flexibilidad de adaptación a distintos sistemas operativos.

El objetivo de TCP/IP es construir una interconexión de redes que proporciona servicios de comunicación universales (Internet). Las comunicaciones entre servicios proporcionan el software y hardware que se ejecuta entre la red física y la aplicación de usuario, y da a éstas aplicaciones una interfaz común [Bautista, 2005].

Para poder interconectar 2 redes necesitamos un dispositivo que esté conectado a ambas redes y que pueda retransmitir paquetes de una a la otra, a este dispositivo se le llama router. Para ser capaz de identificar una computadora en la red a cada una se le asigna una dirección, la dirección IP.

2.2.1. Modelo OSI

Al describir la estructura y función de los protocolos de comunicaciones se suele recurrir a un modelo de arquitectura desarrollado por la OSI (International Standards Organization). Este modelo se denomina Modelo de Referencia OSI (Open Systems Interconnect).

El modelo OSI (mostrado en la Tabla 2.2) permite entender mejor el traspaso de información en una red, ya que explica el funcionamiento a base de siete capas, las cuales tienen funciones determinadas en el proceso de transmisión de datos.

Tabla 2.2: Capas del modelo OSI.

Modelo OSI
Capa de aplicación.
Capa de presentación.
Capa de sesión.
Capa de transporte
Capa de red
Capa de enlace.
Capa física.

Cada capa del modelo representa una función realizada cuando los datos son transferidos. Los niveles superiores de este modelo son más cercanos a la interacción con el usuario,

mientras que las capas inferiores, se relacionan más con la transmisión [Bautista, 2005]. Resumiendo las funciones de cada capa del modelo OSI (Tabla 2.2), se tiene lo siguiente:

- Capa de aplicación: Se entiende directamente con el usuario final, al proporcionarle el servicio de información distribuida para soportar las aplicaciones y administrar las comunicaciones por parte de la capa de presentación.
- Capa de presentación: Permite a la capa de aplicación interpretar el significado de la información que se intercambia. Esta realiza las conversiones de formato en caso de que la información que se transmite o se recibe no estén expresadas en el mismo lenguaje para lograr la comunicación de dispositivos.
- Capa de sesión: Administra la interconexión mediante el suministro de los servicios que se necesitan para establecer la comunicación, flujo de datos y conclusión de la conexión.
- Capa de transporte: Esta capa proporciona el control de extremo a extremo y el intercambio de información con el nivel que requiere el usuario. Representa el corazón de la jerarquía de los protocolos que permite realizar el transporte de los datos en forma segura.
- Capa de red: Proporciona los medios para establecer, mantener y concluir las conexiones entre los sistemas del usuario final. Por lo tanto, la capa de red es la más baja, que se ocupa en la transmisión de extremo a extremo.
- Capa de enlace: Asegura la confiabilidad del medio de transmisión, ya que realiza la verificación de errores, retransmisión y control de flujo de datos que se utilizan en la capa de red.
- Capa física: Se encarga de las características eléctricas, mecánicas y funcionales del procedimiento que se requiere para mover los bits de datos entre cada extremo del enlace de la comunicación.

Basándonos en lo anterior para realizar una aplicación o tarea involucrando el protocolo TCP/IP se debe tener en cuenta las capas del modelo OSI. El modelo OSI, si bien explica la transferencia de datos desde un punto de vista más teórico, en la práctica se usa una

simplificación de éste en cinco capas denominado modelo TCP tal y como se describe en la Tabla 2.3, el cual es más sencillo de entender [Bautista, 2005].

Tabla 2.3: Modelo TCP/IP.

Modelo TCP/IP
Capa de aplicación.
Capa de transporte
Capa de Internet
Capa de acceso

- Capa de aplicación: Se entiende directamente con el usuario final, al proporcionarle el servicio de información distribuida para soportar las aplicaciones.
- Capa de transporte: Se encarga de los servicios de envío de datos con detección y corrección de errores.
- Capa de Internet: Proporciona los servicios básicos de transmisión de paquetes.
- Capa de acceso a la red: Proporciona al sistema los medios para enviar los datos a otros dispositivos conectados a la red.

2.2.2. Protocolo IP

El protocolo IP proporciona un medio para el transporte de paquetes o datos mediante direcciones lógicas IP de 32 bits del origen al destino, sin importar si éstas máquinas están en la misma red, o si hay otras redes entre ellas. IP está implementado en todas las computadoras y routers. Se encarga de la retransmisión de los datos de una computadora a otra, pasando por uno o varios routers nodo a nodo, en caso de ser necesario.

Los datos proporcionados por la capa de transporte son divididos y transmitidos a través de la capa de red (Internet). Durante el camino los datos pueden ser fragmentados en unidades más pequeñas si deben atravesar una red o subred cuyo tamaño de paquete sea más pequeño. En la máquina destino, éstas unidades son reensambladas para volver a tener los datos originales [Bautista, 2005].

2.2.3. Direcciones IP

Para que una red de dos computadoras puedan comunicarse entre sí ellas deben estar identificadas con precisión. TCP/IP utiliza un identificador denominado dirección IP, cuya

longitud es de 32 bytes. La dirección IP identifica tanto a la red a la que pertenece una computadora como a ella misma dentro de dicha red.

Una dirección puede pertenecer a 5 clases diferentes la cual puede ser determinada por la posición del primer 0 en los cuatro primeros bits. Las direcciones están codificadas para permitir una asignación variable de bits para especificar la red y la computadora en la Tabla 2.4 se muestran las clases de direcciones IP [Bautista, 2005].

- Clase A. Pocas redes: hasta 128 redes cada una con muchas computadoras 16, 777, 216.
- Clase B. Un número medio de redes: hasta 16384 redes, cada una con un número medio de computadoras: 65, 536.
- Clase C. Muchas redes: hasta 2, 097, 152 redes, cada una con pocas computadoras: 256.
- Clase D. Se permite transmitir la información a múltiples computadoras al mismo tiempo.
- Clase E. Reservado para el futuro.

Tabla 2.4: Estructura de las direcciones IP.

Direcciones IP.				
32 bits.				
Clase	Datos			Rango de direcciones.
A	0	Red	Host	1.0.0.0-127.255.255.255
B	10	Red	Host	128.0.0.0-191.255.255.255
C	110	Red	Host	192.0.0.0-223.255.255.255
D	1110	Multisesión		224.0.0.0-239.255.255.255
E	11111	Reservado		240.0.0.0-255.255.255.255

Las direcciones IP clase A son reservadas para los gobiernos, las direcciones IP clase B se otorgan a medianas empresas o universidades, los usuarios normales usarán direcciones de clase C y las direcciones del tipo clase d y e se utiliza principalmente en redes de investigación. La Universidad Michoacana esta dentro de la clase B.

2.2.4. Protocolo TCP

El protocolo TCP permite una comunicación fiable entre dos aplicaciones. De esta forma, las aplicaciones que lo utilicen no tienen que preocuparse de la integridad de la

información. Durante una comunicación usando el protocolo TCP, las dos aplicaciones deben establecer una conexión. La máquina emisora (la que solicita la conexión) se llama cliente, y la máquina receptora se llama servidor. Por eso es que decimos que estamos en un entorno cliente-servidor. Esta conexión garantiza que todos los datos lleguen correctamente de forma ordenada y sin duplicados.

La unidad de datos del protocolo es el byte, de tal forma que la aplicación origen envía bytes y la aplicación destino recibe estos bytes. Sin embargo, cada byte no se envía inmediatamente después de ser generado por la aplicación, sino que se espera a que haya una cierta cantidad de bytes, se agrupan en un segmento y se envía el segmento completo. Para ello son necesarias unas memorias intermedias o buffers.

Cada vez que se abre una conexión, se crea un canal de comunicación bidireccional en el que ambas aplicaciones pueden enviar y recibir información, es decir, una conexión es Full Duplex.

Para posibilitar la comunicación y que funcionen bien todos los controles que la acompañan, los datos se agrupan, es decir, se agrega un encabezado a los paquetes de datos que permitirán sincronizar las transmisiones y garantizar su recepción.

TCP permite colocar los paquetes o datos nuevamente en orden cuando vienen del protocolo IP, monitorear el flujo de datos y así evitar la saturación de la red, segmentar los datos para entregarlos al protocolo IP, multiplexar los datos, es decir que la información que viene de diferentes fuentes puedan circular en la misma línea simultáneamente y permite comenzar y finalizar la comunicación.

TCP posibilita la realización de una tarea importante: multiplexar o demultiplexar, es decir, transmitir datos desde diversas aplicaciones en la misma línea o, en otras palabras, ordenar la información que llega en paralelo. Estas operaciones se realizan empleando el concepto de puertos (o conexiones), es decir, un número vinculado a un tipo de aplicación que cuando se combina con una dirección de IP, permite determinar en forma exclusiva una aplicación que se ejecuta en una máquina determinada [Palominos, 2006].

2.2.5. Protocolo TELNET

TELNET es un protocolo de red usado en Internet para proveer una comunicación bidireccional entre 2 o más computadoras. TELNET es anterior a Internet y se ejecutaba de ordenador a ordenador, es decir por medio de llamada directa de la computadora

cliente a la computadora servidor, TELNET es ejecutado dentro de Internet, mediante el protocolo TCP/IP.

Internet ofrece, además de sus servicios más populares (email, navegación) otros servicios como TELNET, antes de que la WWW (World Wide Web, en informática, la World Wide Web es un sistema de distribución de información basado en hipertexto o hipermedios enlazados y accesibles a través de Internet.) hiciera posible la navegación gráfica a través de páginas de Internet, los ordenadores conectados a internet solo entendían secuencias de comandos. La llegada de la navegación gráfica ha relegado a TELNET a usos más restringidos, normalmente relacionados con la administración remota de equipos como va ser el caso de esta tesis.

Al realizar una conexión TELNET entre dos computadoras podemos estar operando desde nuestra casa como si nos encontrásemos en nuestra computadora del trabajo, accediendo a sus mismos recursos, según los permisos con los que contemos los cuales son programados. Con TELNET podemos acceder a miles de ordenadores remotos, estableciendo conexión directa con ellos. En la actualidad lo normal es que las conexiones TELNET se hagan vía Internet [Bautista, 2005].

2.3. Bases teóricas del Sistema Global para las Comunicaciones Móviles

2.3.1. Antecedentes del estándar GSM

A principios de la década de los 80, los sistemas de telefonía móvil analógicos existentes tenían limitaciones. Primero, la potencial demanda de servicios móviles fue mayor de la capacidad que las existentes redes analógicas podían soportar. Segundo, las diferentes formas de operación no ofrecían compatibilidad para los usuarios de móviles: una terminal bajo el sistema TACS (Total Access Communication System, servicio de telefonía móvil analógico puesto en funcionamiento en el Reino Unido en 1985) no podía acceder dentro de una red NMT (sistema de telefonía móvil analógico de cobertura escandinava), y viceversa. Además, el diseño de un nuevo sistema de telefonía celular requería una cantidad de investigación que ningún país europeo podía afrontar de forma individual. Todas estas circunstancias apuntaron hacia el diseño de un nuevo sistema.

El principal requisito para un sistema de radio común, es el ancho de banda de radio. Esta condición había sido ya prevista unos pocos años antes, en 1978, cuando se decidió reservar la banda de frecuencia de 900 ± 25 MHz para comunicaciones móviles en Europa. Todos estos factores, llevaron a la creación en 1982 de un nuevo cuerpo de estandarización cuya tarea era especificar un único sistema de radiocomunicaciones para Europa a 900 MHz. En 1990, por requerimiento del Reino Unido, se añadió al grupo de estandarización la especificación de una versión de GSM a la banda de frecuencia de 1800 ± 75 MHz. A esta variante se le llamó DCS1800 (Digital Cellular System 1800) y se convirtió en un estándar internacional llamado Sistema Global para las Comunicaciones Móviles (GSM, Global System for Mobile Communications), cambiando de esta forma el significado actual de las siglas GSM [Rosario, 2010].

2.3.2. Estándar GSM

El estándar GSM es el sistema digital celular. Se trata de un estándar internacional para las comunicaciones radio digitales creado por el Instituto Europeo de Estándares de Telecomunicaciones (ETSI, European Telecommunications Standards Institute).

El estándar GSM ha sido desarrollado para permitir las comunicaciones móviles en todos los mercados y empezó a utilizarse en el año 1991. En Europa se aplica en dos bandas de frecuencia, a 900 MHz y a 1800 MHz (GSM 900 y GSM 1800 respectivamente), mientras que en América se aplica en la banda de 1900 MHz (GSM 1900) y en el Sureste Asiático y Japón a 850 MHz (GSM 850).

Actualmente, el estándar GSM ocupa más del 85 % del mercado de las telecomunicaciones Inalámbricas.



Figura 2.2: Diagrama de bloques simplificado para una comunicación a través de la red GSM.

En una red GSM se distinguen 3 bloques principales: Transmisor, Estación fija y Receptor. En la Figura 2.2 se muestra un diagrama de bloques general de un sistema de comunicaciones a través de la red GSM [Cser, 2006].

2.3.3. Comunicación a través de la red GSM.

En la red GSM el transmisor y receptor también son llamados estaciones móviles, el cual será el encargado de convertir la información de la fuente de información (voz, datos, vídeo, etc.), en una forma adecuada para su transmisión y recepción [Cser, 2006].

En una red GSM, la estación móvil es utilizada para realizar y recibir llamadas, además brindará al usuario una interfaz la cual puede incluir micrófono, pantalla, teclado, etc. y también una interfaz para otro dispositivo tal como una computadora o un micro-controlador. La estación móvil está compuesta por dos módulos los cuales son:

- El módulo de identidad del subscriptor (SIM, Subscriber Identity Module) el cual es una tarjeta inteligente que contiene toda la información acerca del subscriptor.
- El equipo móvil (ME, Mobile Equipment) que contiene todas las herramientas básicas para acceder a la red.

En el sistema GSM, entre el transmisor y receptor existe un bloque llamado estación fija, el cual es el encargado de lograr que la comunicación entre dos o más puntos se lleve a cabo de la manera adecuada, este bloque está formado por tres subsistemas que son:

- El subsistema de estación base (BSS, Base Station Sub-System). Su función es controlar las estaciones fijas (transmisoras/receptoras) y darle servicio a los dispositivos móviles para mantener la comunicación entre los usuarios. Este subsistema se compone de dos módulos:
 - ▷ La estación base transmisora/receptora (BTS, Base Transceiver Station). La BTS implementa la interfaz de comunicación de todas las estaciones móviles que se localizan dentro de su área de cobertura. Esto incluye la modulación/demodulación de señales, la sincronización de las mismas y la codificación de errores.
 - ▷ Controlador de las estaciones base (BSC, Base Station Controller). El BSC constituye un conjunto de disposiciones para manejar las conexiones de las BTS's que están bajo su control, normalmente un BSC controla alrededor de 70 bases transmisoras/receptoras.

- El subsistema de red (NSS, Network Sub-System). Su función es verificar la ubicación de las llamadas; desde cual equipo móvil son realizadas y el encaminamiento de ellas. Este subsistema está compuesto por 3 módulos:
 - ▷ El centro de conmutación móvil (MSC, Mobile Switching Centre). Las funciones del MSC son: el encaminamiento de llamadas hacia las redes externas públicas o privadas de telefonía o datos, controlar los servicios suplementarios y recopilar la información necesaria para tarificación.
 - ▷ El registro de localización del visitante (VLR, Visitor Location Register). La función principal que desempeña el VLR es la de dar una copia local de la información del usuario para manejar llamadas.
 - ▷ El registro de localización local (HLR, Home Location Register). Es el encargado de almacenar información específica de cada subscriptor, además, este registro contendrá detalles del área MSC en la cual el usuario está registrado y los servicios a los cuales tiene acceso. Una HLR tiene la capacidad para manejar la información de cientos de miles de usuarios. La información que está contenida en el HLR debe ser accesada utilizando la identidad de subscriptor móvil internacional (IMSI, International Mobile Subscriber Identity) o desde el número de estación móvil internacional ISDN (MSISDN, Mobile Station Inter ISDN), el cual es esencialmente el número telefónico del usuario.
- El subsistema de operación (OSS, Operation Sub-System). Este es muy importante para el operador de red para identificar problemas en la red en una etapa temprana y corregir el problema rápida y eficientemente. Esto además es importante para el operador de red para ser capaz de hacer cambios en la configuración de red con un mínimo de esfuerzo y sin afectar el servicio que provee a sus subscriptores. Los bloques funcionales asociados con el subsistema de operación del GSM son:
 - ▷ Centro de operación y mantenimiento (OMC, Operations and Maintenance Centre). El objetivo del OMC es brindar al cliente un soporte rentable para el funcionamiento local y regional ya que contempla las actividades de mantenimiento para el buen funcionamiento de la red GSM. En resumen, las funciones de red llevadas a cabo por el OMC son:

- Soporte para el mantenimiento.
 - Administrador de errores.
 - Administrador de rendimiento.
 - Versión de software y control de configuración.
 - Estado de la red.
- Colección de tráfico de la red.
- ▷ Centro de manejo de red (NMC, Network Management Centre). El NMC provee un control global y centralizado sobre las operaciones y mantenimiento de las redes. Estas tareas son soportadas por los OMC's haciéndolos responsables de la administración de la red regional. El OMC es el administrador central regional mientras que el NMC es el administrador central global.
- ▷ Centro de administración (ADC, Administration Centre). El ADC es el encargado de las funciones administrativas de la red, tales como: la cuenta del usuario y el cobro de las llamadas.

2.3.4. Servicios del GSM

Los servicios que brinda el sistema GSM se agrupan en tres categorías [Cser, 2006]:

- Teleservicios. Los teleservicios cubren la telefonía regular (como lo son las transmisión o recepción de llamadas), llamadas de emergencia y mensajes de voz. El servicio más importante que provee el GSM es la telefonía. Este servicio posibilita llamadas en forma bidireccional entre usuarios de la red GSM y cualquier subscritor de teléfono que esté disponible a través de la red general de telefonía: los subscritores de teléfonos fijos alrededor del mundo y los subscritores de la red móvil u otro subscritor de alguna red específica que pueda conectarse a la red de telefonía pública pueden estar disponibles. Los servicios a los que se tiene acceso con el modulo SM5100B son mensajes SMS conexión GSM/GPRS y conexión TCP/IP.
- Servicios portadores. Los servicios portadores deben de ser proporcionados a través de un hardware adicional. Cuando algún usuario quiere hacer uso de un servicio portador deberá conectar un hardware adicional a la estación móvil en cada una de

las estaciones móviles en uso; este hardware adicional puede ser una computadora. A la unión de la estación móvil y hardware se le conoce como equipo terminal de datos (DTE, Data Terminal Equipment). Los servicios portadores proveen sus propios protocolos de comunicaciones y el GSM solo provee el método de transporte de información.

- Servicios adicionales. Los servicios adicionales reciben este nombre debido a que se trata de características adicionales al servicio de transportes y teleservicios. Los principales servicios adicionales son: llamada restringida, desvío de llamadas y la identificación de una llamada entrante.

2.3.5. Servicio de Mensajes Cortos (SMS)

Servicio de Mensajes Cortos (SMS) es un servicio inalámbrico aceptado globalmente este permite la transmisión de mensajes alfanuméricos entre clientes de teléfonos móviles y sistemas externos tales como correo electrónico, y sistemas de mensajes de voz.

SMS apareció en 1991 en Europa, SMS estuvo disponible en las primeras redes inalámbricas digitales construidas por BellSouth Mobility y Nextel. En 1998, con el desarrollo de las redes basadas en GSM se ayudo a la completa implementación del SMS. El SMS punto a punto provee un mecanismo para transmitir mensajes cortos de y hacia equipos Móviles (Celulares). Tras el envío de un mensaje, éste no sigue directamente para el destinatario sino para un centro de mensajes (SMSC), que lo almacena y envía posteriormente. Este centro hace también la cobranza posterior del servicio. El centro de mensajes encamina después el mensaje para el destinatario, cuando el móvil esté conectado a la red. De esta manera es posible tener la certeza que el mensaje llegó a su destino, porque el centro de mensajes puede notificar el remitente en caso de que la operación falle. Una característica del servicio es que en un equipo Móvil activo es capaz de recibir o enviar un mensaje corto en cualquier momento, independiente si hay o no una llamada de voz o datos en progreso. SMS también garantiza la entrega de los mensajes cortos por la red. Errores temporales son identificados y el mensaje es guardado en la red hasta que el destino esté disponible.

SMS está caracterizado por entrega de paquetes con un bajo uso del ancho de banda para la transferencia de mensajes. Las primeras aplicaciones de SMS estaban enfocadas en eliminar los sistemas de búsqueda de personas que solamente permitían una comunicación unidireccional, SMS permite mensajería en las dos direcciones y servicios de notificación

principalmente mensajes de voz. Al madurar esta tecnología y las redes se fueron agregando, una variedad de servicios como el correo electrónico, fax, servicios de búsqueda, bancos interactivos y servicios de información y aplicaciones inalámbricas tales como el modulo de identidad del subscritor (SIM).

Los beneficios del SMS a los clientes se centran en la conveniencia, flexibilidad y la integración de servicios de mensajes y acceso a datos. Desde esta perspectiva, el beneficio es ser capaz de usar un equipo móvil como una extensión del computador [Cser, 2006].

2.3.5.1. Elementos y arquitectura necesaria para el servicio SMS

Los elementos de red necesarios para proveer el servicio SMS [Cser, 2006] (ver Figura 2.3.) son:

- Las Entidades de Mensajería Corta (Short Messaging Entities - SME): Es una entidad que puede enviar o recibir mensajes cortos. Puede ser localizada en la red fija, la estación móvil u otro centro de servicio.
- El Centro de Servicio de Mensaje Corto (Short Message Service Center - SMSC) es el responsable de la transmisión, almacenamiento y envío de mensajes cortos entre el SME y la estación móvil.
- El Centro de Conmutación Móvil SMS (SMS Gateway/Interworking Mobile Switching Center - SMS GMSC) es un centro de conmutación de mensajes encargado de recibir el mensaje del SMSC y entregarlo al MSC que da servicio a la estación móvil.
- Registro de Localización Local. (Home Location Register - HLR): Es la base de datos para el almacenamiento permanente y manejo de perfiles de servicio y subscripciones. El HLR provee la información de encaminamiento hacia el cliente indicado, también informa al SMSC del intento de entrega de un mensaje corto a una estación móvil que ha resultado fallido.
- Registro de Localización del Visitante (Visitor Location Register - VLR): El VLR es la base de datos que contiene la información temporal acerca de los clientes. Esta información se necesita por el MSC (Mobile Switching Center - MSC) que ejecuta las funciones de conmutación del sistema y las llamadas de control hacia y desde otros teléfonos o sistemas de datos.

- Estación Base del sistema. (Base Station System - BSS): Todas las funciones relacionadas con la radio se ejecutan en la BSS, la cual consiste en unos controladores de estación base (Base Station Controllers - BSCs) y estaciones base transceptoras (Base Transceiver Stations - BTSs) que se encargan de transmitir la voz y el tráfico de datos entre las estaciones móviles.
- La Estación Móvil (Mobile Station - MS): Es el terminal inalámbrico capaz de recibir y originar mensajes cortos, así como llamadas.

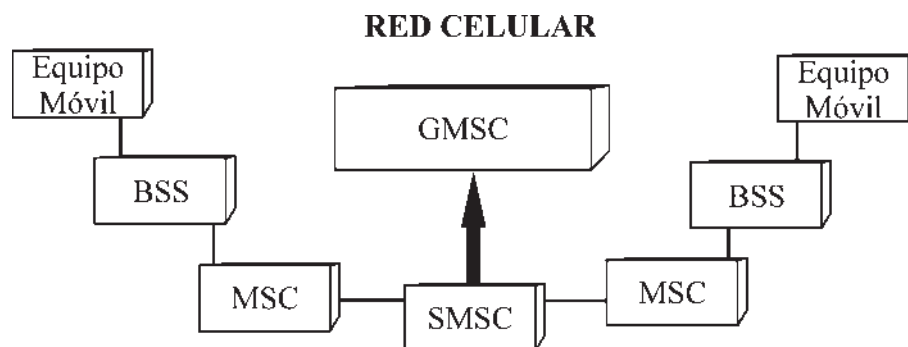


Figura 2.3: Elementos de red GSM necesarios para proveer el servicio SMS.

2.3.5.2. Elementos de señalización

SMS hace uso de la Parte de Aplicación Móvil (Mobile Application Part - MAP), la cual define los métodos y mecanismos de comunicación en redes inalámbricas.

Las siguientes operaciones MAP son necesarias:

Una vez que un mensaje se envía, es recibido por el Centro de Servicio de Mensaje Corto (SMSC) que debe enviarlo entonces al dispositivo móvil apropiado. Para hacer esto, el SMSC envía un requerimiento de SMS al registro de localización (HLR) para encontrar al cliente. Una vez el HLR recibe la llamada, responde al SMSC informando el estado del cliente: inactivo o activo (el estado activo es cuando el cliente se encuentra conectado a la red GSM y el estado inactivo se refiere a cuando no se encuentra dentro de la red) y en dónde está. Si la respuesta es "inactivo", entonces el SMSC almacenará el mensaje por un periodo de tiempo y cuando el cliente activa su dispositivo, el HLR envía una notificación de SMS al SMSC, que intentará la entrega. Si el estado es "activo", entonces el sistema llama al dispositivo, y si responde, el mensaje se entrega. El SMSC recibe la verificación de que el mensaje se recibió por el usuario terminal y etiqueta el mensaje como "enviado" para no tratar de enviarlo de nuevo [Rosario, 2010].

2.4. Introducción al protocolo ZigBee

ZigBee es un protocolo de comunicaciones inalámbrico basado en el estándar de comunicaciones para redes inalámbricas IEEE_802.15.4 de redes inalámbricas de área personal (Wireless Personal Area Network, WPAN). Creado por ZigBee Alliance, una organización, de más de 200 grandes empresas (Mitsubishi, Honeywell, Philips, entre otras), muchas de ellas fabricantes de semiconductores. ZigBee permite que dispositivos electrónicos de bajo consumo puedan realizar sus comunicaciones inalámbricas.

Las comunicaciones ZigBee se realizan en la banda libre de 2.4GHz. Este protocolo realiza las comunicaciones a través de una única frecuencia, es decir, de un canal. Normalmente puede escogerse un canal de entre 16 posibles. El alcance depende de la potencia de transmisión del dispositivo así como también del tipo de antenas utilizadas (cerámicas, dipolos, etc). El alcance normal con antena dipolo en línea vista es de aproximadamente (tomando como ejemplo el caso de Maxstream, en la versión de 1mW de potencia) de 100m y en interiores de unos 30m. La velocidad de transmisión de datos de una red ZigBee es de hasta 256kbps. Una red ZigBee la pueden formar, teóricamente, hasta 65535 equipos, es decir, el protocolo está preparado para poder controlar en la misma red ésta cantidad enorme de dispositivos.

El uso del protocolo ZigBee va desde reemplazar un cable por una comunicación serial inalámbrico, hasta el desarrollo de configuraciones punto a punto, multipunto o redes complejas de sensores [Manual Xbee®(R), 2007].

Una red ZigBee la forman 3 tipos de elementos. Un dispositivo Coordinador, dispositivos routers y dispositivos finales (endpoints).

- Coordinador. Este tiene la única función de formar una red. Se encarga de establecer el canal de comunicaciones y del PAN ID (identificador de red) para toda la red. Una vez establecidos estos parámetros, el coordinador puede formar una red, permitiendo unirse a él a dispositivos routers y endpoints. Una vez formada la red, el coordinador hace las funciones de router, esto es, participa en el enrutado de paquetes y ser origen y destinatario de información.
- Routers. Los dispositivos que son configurados como router crean y mantienen información sobre la red para determinar la mejor ruta para transmitir un paquete de información. Un router debe unirse a una red ZigBee antes de poder actuar como

router retransmitiendo paquetes de otros routers o de endpoints.

- Enddevice. No tienen capacidad de enrutar paquetes. Son los que interactúan siempre a través de su nodo padre, ya sea este un coordinador o un router, es decir, no puede enviar información directamente a otro enddevice. Normalmente estos equipos van alimentados a baterías. El consumo es menor al no tener que realizar funciones de enrutamiento.

Cada módulo ZigBee, tiene una dirección única. En el caso de los módulos ZigBee cada uno de ellos tiene una dirección única de 64 bits que viene grabada de fábrica. Por otro lado, la red ZigBee, utiliza para sus algoritmos de ruteo direcciones de 16 bits. Es decir cada vez que un dispositivo se asocia a una red ZigBee, el coordinador al cual se asocia le asigna una dirección única en toda la red de 16 bits. Por eso el número máximo teórico de elementos que puede haber en una red ZigBee es de $2^{16} = 65535$, número máximo de direcciones de red que se pueden asignar [Manual Xbee®[®], 2007].

2.4.1. Modos de operación

Los módulos Xbee®[®], pueden operar en los siguientes 5 modos [Manual Xbee®[®], 2007].

2.4.1.1. Modo RECIBIR/TRANSMITIR

Este modo de operación se tiene cuando el módulo recibe algún paquete RF (radio frecuencia) a través de la antena (modo Recibe) o cuando se envía información serial al buffer del pin 3 (Data in) que luego será transmitida (modo Transmit) [Manual Xbee®[®], 2007].

La información transmitida puede ser Directa o Indirecta.

- Modo Directo: la información se envía inmediatamente a la dirección de destino.
- Modo Indirecto: la información es retenida por el módulo durante un período de tiempo y es enviada sólo cuando la dirección de destino la solicita.

Se puede enviar información por dos formas diferentes: Unicast y Broadcast.

- Modo Unicast: la comunicación es desde un punto a otro, y es el único modo que permite respuesta de quien recibe el paquete RF, es decir, quien recibe debe enviar un bit llamado ACK (paquete llamado así, y que indica que recibió el paquete, el

usuario no puede verlo, es interno de los módulos) de la dirección de origen. Quien envió el paquete, espera recibir un ACK, en caso de que no le llegue, reenviará el paquete hasta que se reciba el ACK.

- Modo Broadcast: la comunicación es entre un nodo y a todos los nodos de la red. En este modo, no hay confirmación por ACK al momento de la transmisión.

2.4.1.2. Modo de bajo consumo (SleepMode)

El modo de Bajo Consumo o modo Sleep, corresponde a cuando el módulo entra en un estado de bajo consumo de energía. Esto depende de la configuración en la que se encuentra [Manual Xbee®[®], 2007].

2.4.1.3. Modo de comando

Este modo permite ingresar comandos AT al módulo Xbee®[®], para configurar, ajustar o modificar parámetros del módulo [Manual Xbee®[®], 2007].

2.4.1.4. Modo transparente

En este modo todo lo que ingresa como paquete RF, es guardado en el buffer de salida y luego enviado. El modo Transparente viene por defecto en los módulos Xbee®[®]. Este modo de operación está destinado a la comunicación punto a punto, donde no es necesario ningún tipo de control. Se utiliza cuando se necesita reemplazar alguna conexión serial por cable, ya que es la configuración más sencilla posible [Manual Xbee®[®], 2007].

2.4.1.5. Modo de operación API

Este modo es más complejo, pero permite el uso de frames (paquetes de datos) con cabeceras que aseguran la transmisión de los datos. Extiende el nivel en el cual la aplicación del cliente, puede interactuar con las capacidades de red del módulo.

Cuando el módulo Xbee®[®] se encuentra en este modo, toda la información que entra y sale, es empaquetada en frames, que definen operaciones y eventos dentro del módulo.

El modo API, permite la configuración del módulo y ruteo de la información. Un cliente puede enviar información al módulo Xbee®[®]. Estos datos serán contenidos en un frame cuya cabecera tendrá información útil referente al módulo [Manual Xbee®[®], 2007].

2.4.1.6. Modo de operación IDLE

Cuando el módulo no se está en ninguno de los modos descritos, se encuentra en éste. Es decir, si no está ni transmitiendo ni recibiendo, ni ahorrando energía ni en el modo de comandos, entonces se dice que se encuentra en un estado al que se le llama IDLE [Manual Xbee®[®], 2007].

2.4.2. Direccionamiento de los módulos

Los módulos permiten 2 tipos de direccionamiento. La de 16 bits y la de 64 bits. La principal diferencia es que en la de 64 bits, es posible obtener una mayor cantidad de direcciones y por lo tanto, una mayor cantidad de nodos o equipos funcionando en la misma red. Son a través de estas direcciones que los módulos se comunican entre sí [Manual Xbee®[®], 2007].

2.4.3. Modos de conexión

Los módulos ZigBee tienen 4 formas de conexión.

2.4.3.1. Modo de conexión transparente

Esta es la conexión que viene por defecto y es la forma más sencilla de configurar el módem. Básicamente todo lo que pasa por el puerto UART (comunicación RS232 a un nivel de voltaje de 3.3 volts), es enviado al módulo deseado, y lo recibido en el módulo, es enviado devuelta por el mismo puerto UART [Manual Xbee®[®], 2007].

2.4.3.1.1. Modo de conexión punto a punto En esta configuración se reemplaza la comunicación serial por un cable. La idea, es definir arbitrariamente una dirección para un módulo, el cual se va a comunicar con otro del cual se sabe su dirección dentro de la red, las direcciones de ambos módulos son definidas arbitrariamente. Como se muestra en la Figura 2.4 cada módulo define su dirección, y escribe la dirección del módulo al cual se desea conectar, también se define la red en la que se encuentran los 2 dispositivos (debe ser forzosamente la misma) [Manual Xbee®[®], 2007].

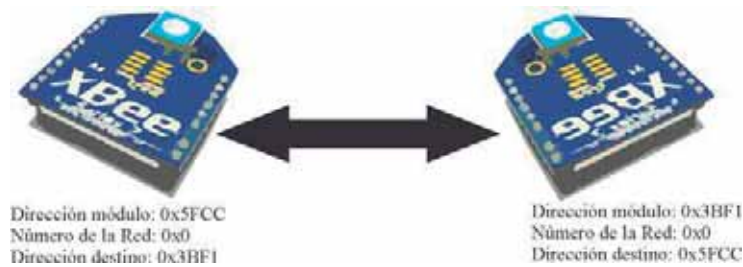


Figura 2.4: Configuración para la operación punto a punto en los módulos Xbee®.

2.4.3.1.2. Modo de conexión punto a multipunto Este modo de conexión permite transmitir información, desde la entrada serial de un módulo a uno o varios módulos conectados a la misma red de manera más controlada, ya que se necesitan las direcciones de los otros módulos, por lo que existe mayor seguridad.

En la Figura 2.5 se muestra la idea general de una conexión punto a multipunto.

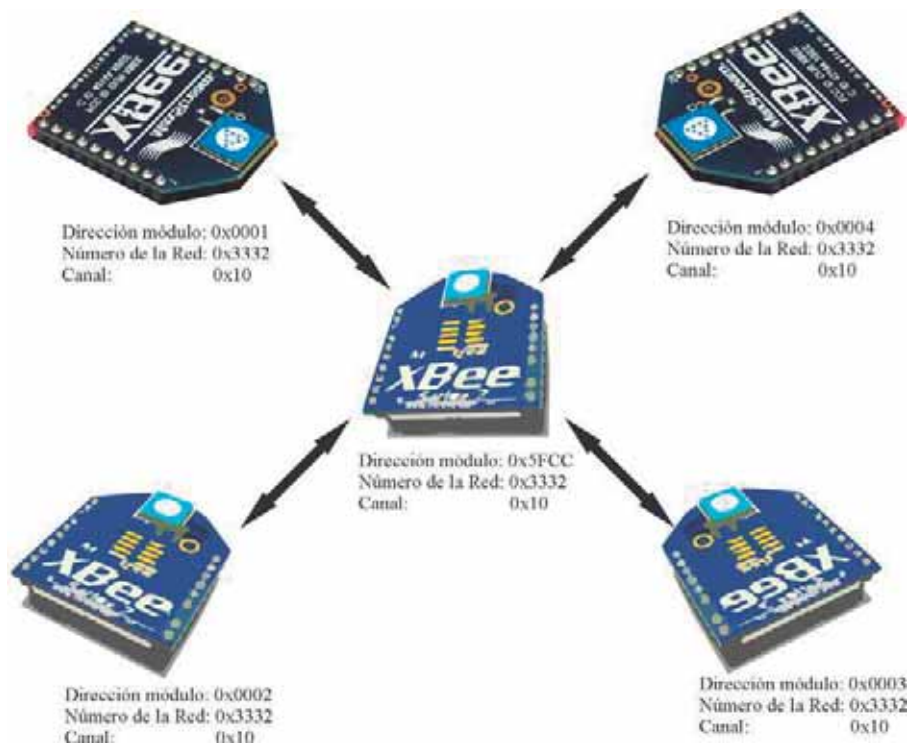


Figura 2.5: Configuración para la operación punto a multipunto en los módulos Xbee®.

Para la conexión punto a multipunto cada nodo se configura con una dirección propia distinta para cada módulo, pero utilizando el mismo canal y el mismo número de red. En la Figura 2.6 se puede observar que se hace la selección del canal por el cual se van a comunicar los módulos, se disponen de 16 canales según el protocolo IEEE 802.15.4. Como se observa en la Figura 2.6 éste estándar indica que entre cada canal, deben existir

5 MHz de diferencia, partiendo de la frecuencia central 2.405 GHz, se llegan hasta los 2.480 GHz.

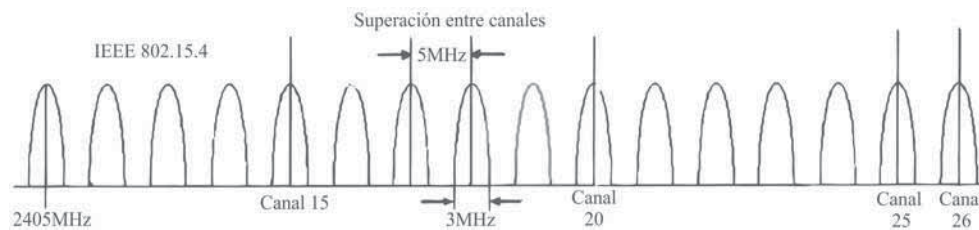


Figura 2.6: Ancho de banda de los canales de frecuencia existentes en los módulos Xbee ®.

Se observa que hay 16 canales disponibles, sin embargo, los valores se asignan desde el 11 hasta el 26.

La Tabla 2.5 muestra la frecuencia central de cada canal, así como su límite inferior y superior:

Tabla 2.5: Canales de frecuencia a los que se puede tener acceso en los módulos Xbee ®.

Frecuencia GHz				
Canal	Hexadecimal	Frecuencia Inferior	Frecuencia Central	Frecuencia Superior
11	0x0B	2.4025	2.4050	2.4075
12	0x0C	2.4075	2.4100	2.4125
13	0x0D	2.4125	2.4150	2.4175
14	0x0E	2.4175	2.4200	2.4225
15	0x0F	2.4225	2.4250	2.4275
16	0x10	2.4275	2.4300	2.4325
17	0x11	2.4325	2.4350	2.4375
18	0x12	2.4375	2.4400	2.4425
19	0x13	2.4425	2.4450	2.4475
20	0x14	2.4475	2.4500	2.4525
21	0x15	2.4525	2.4550	2.4575
22	0x16	2.4575	2.4600	2.4625
23	0x17	2.4625	2.4650	2.4675
24	0x18	2.4675	2.4700	2.4725
25	0x19	2.4725	2.4750	2.4775
26	0x1A	2.4775	2.4800	2.4825
Frecuencia base 2.405GHz.				

La elección del canal debe ser cuidadosa, ya que otras tecnologías como WI-FI o Bluetooth utilizan el mismo espectro de frecuencias, por lo que se podría producir interferencia [Manual Xbee®, 2007].

2.4.3.2. Modo de conexión broadcast

Esta configuración nos permite el envío de información desde un nodo a varios nodos en una misma red. La información recibida es la misma para todos los nodos. Para que un módulo entregue datos a todos los nodos, es necesario ajustarlo con la dirección de Broadcast la cual es indicada por el fabricante. Cualquier módulo que reciba un paquete con una dirección de destino de Broadcast será aceptado. La Figura 2.7 muestra un esquema general de una red de Broadcast. Para la conexión punto a multipunto cada nodo se configura con una dirección propia distinta para cada módulo, el mismo canal y el mismo número de red, pero con una dirección destino igual para todos los módulos la dirección destino, esta forma de conexión está dada por una parte alta dada por el numero hexadecimal 0x00000000 y una parte baja dada por el numero hexadecimal 0x0000FFFF [Manual Xbee®(R), 2007].



Figura 2.7: Configuración Broadcast en los módulos Xbee®.

2.4.3.3. Modo de conexión cable virtual I/O

Con ésta configuración se crean los llamados Cables Virtuales. Se utilizan para crear un canal de comunicación de manera transparente entre los pines de un módulo y otro.

Cada pin de entrada tiene su propio pin de salida ya definido entre nodos, (esto permite una forma totalmente simple de enviar información, controlar o medir de manera sencilla y rápida, sin necesidad de complicadas configuraciones). El esquemático de los pines se muestra en la Figura 2.8.



Figura 2.8: Vista superior del diagrama de pines del módulo Xbee®.

Para crear un cable virtual se debe entender que los pines de entrada /salida (I/O) de cada modulo ZigBee, están asociados entre sí es decir, vienen en pares. Por esto, sólo se pueden crear cables entre estos pares de pines de distintos módulos, la información que vea cada entrada o salida de un módulo su par en cable virtual vera el mismo dato [Manual Xbee®, 2007].

2.4.4. Modo de conexión NonBeacon. Peer toPeer

Una red peer-to-peer permite que todos los módulos, se conecten con todos, es decir, se crea una conexión de par en par con cada uno de los módulos de la red. El modo de conexión NonBeacon es la configuración por defecto y permite establecer una red peer-to-peer donde cada módulo puede hacer las funciones de maestro o esclavo.

La configuración de red Non-Beacon, se refiere a que cada nodo se mantiene despierto siempre. Por lo que los demás dispositivos que se conectan a él, pueden entrar en modo Sleep (ahorro de energía), y sólo despertarse cuando sea necesario para enviar datos. En una red Beacon, los dispositivos enrutadores están siempre en modo Sleep, y envían señales de su existencia (llamadas Beacon) cada cierto intervalo al resto de la red. Así para poder comunicarse, deben estar totalmente organizados todos los dispositivos, ya que de no ser así, existe la posibilidad de perderse la señal Beacon y no poder enviar hasta la próxima entrega. La ventaja de las redes Beacon, es el ahorro de energía. Por este motivo las redes Non-Beacon están pensadas para dispositivos que posean una alimentación segura,

mientras las Beacon, para alimentación autónoma, como baterías [Manual Xbee® , 2007].

2.4.5. Conexión NonBeacon con coordinador

Es básicamente lo mismo que una red punto-multipunto, con la diferencia de que existe un módulo central que posee ciertas propiedades y características que le permiten administrar la red. En esta red, el módulo central es llamado Coordinador, mientras que el resto de módulos son llamados Dispositivos Terminales (enddevice). Un mismo módulo ZigBee puede ser configurado para funcionar como Coordinador o como Dispositivo Terminal [Manual Xbee® , 2007].

2.4.6. Conexión API

Esta conexión, agrega información extra a los paquetes de datos RF. Ya no son enviados de forma transparente, sino que cada paquete de datos, son almacenados dentro de un frame, con una estructura definida que permite una forma más robusta para enviar datos. Esto permite entre otras cosas determinar el origen de algún paquete recibido dentro de la red. Cuando la configuración API está activada, cada paquete RF que se envía o recibe se encapsula en un frame de datos UART. El frame se observa en la Figura 2.9.

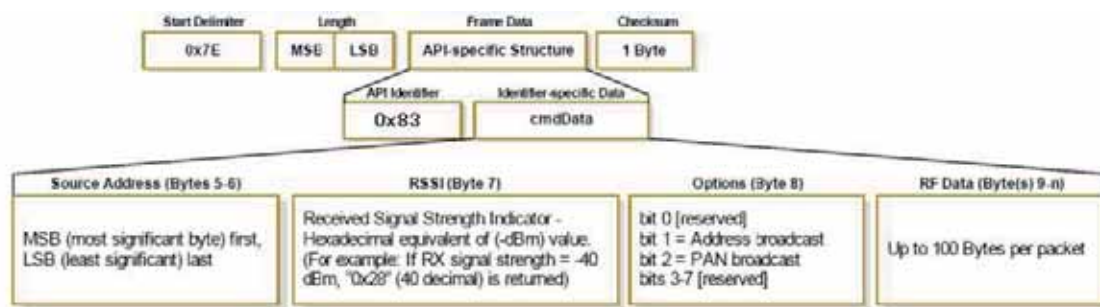


Figura 2.9: Estructura del frame del modo API en los módulos Xbee®.

El primer byte 0x7E indica el comienzo del frame. Los dos bytes siguientes indican el largo solamente del Frame de Datos (Frame Data) (es decir, todo el frame sin contar ni el byte 0x7E, ni el largo-Length, ni el byte Checksum). La estructura API que viene después se compone de lo siguiente: El identificador API 0x83 (cuarto byte del frame) indica que se están recibiendo datos utilizando direccionamiento de 16 bits. Los dos bytes siguientes (el byte 5 y 6) indican la dirección de origen. El byte 7 indica el RSSI que permite determinar la potencia de la señal desde donde vienen los datos. El byte 8, se divide en bit, de los

cuales el bit 1 indica si es un Broadcast de direccionamiento utilizando la misma PAN y el bit 2 que indica un Broadcast de todas las redes PAN. Luego a partir del byte 9, hasta el número de byte dado por el byte de Length, corresponden a los datos que fueron transmitidos [Manual Xbee®[®], 2007].

Capítulo 3

Descripción del Sistema de Nivel de Líquido

Uno de los principales sistemas de la ingeniería de control son los correspondientes a un sistema de nivel de líquido. La mayoría de los sistemas de nivel de líquido son utilizados en diversos procesos, algunos de los más importantes son: embotellado, fermentación, producción de alimentos, refinado de azúcar, generación de energía eléctrica, etc.

El control del nivel de líquido en depósitos y el flujo entre ellos es un problema básico en los procesos industriales. Los procesos industriales requieren líquidos para ser bombeados, almacenados en tanques y luego bombeados a otro tanque. Muchas veces el líquido será procesado por tratamientos químicos en los tanques, pero siempre el nivel de fluido en los tanques debe ser controlado y el flujo entre ellos también.

En los procesos industriales tales como la industria petroquímica, la fabricación de papel y las de tratamiento de agua donde los sistemas hidráulicos son una parte fundamental, una de las variables más importantes a controlar es el nivel de líquido de los tanques del sistema. En algunas ocasiones, otra de las variables que le sigue en importancia en este tipo de sistemas es el control de la temperatura del depósito de agua y el caudal (de entrada o salida). En estos procesos industriales en ocasiones el líquido es procesado por tratamientos químicos en los tanques, pero siempre el nivel de líquido y el flujo de agua de los tanques debe ser controlado [Barajas, 2010].

3.1. Descripción del sistema de nivel de líquido

El Sistema de nivel de líquido empleado se muestra en la Figura. 3.1. Este sistema está compuesto por dos tanques para almacenamiento de fluidos de 38.6x38.6x58cm los cuales se pueden observar en la Figura. 3.2, con aproximadamente 74 litros de capacidad, un depósito de agua con una capacidad de 100 litros (Figura 3.3), estos están alimentados por una bomba de agua de 127 Vca monofásica como se observa en la Figura 3.4, los tanques están interconectados a través de dos electroválvulas proporcionales industriales de marca Honeywell (Figura 3.5), y para determinar el nivel de líquido, se utilizan sensores ultrasónicos de la marca Banner (Q45U), los cuales se muestran en la Figura 3.6.

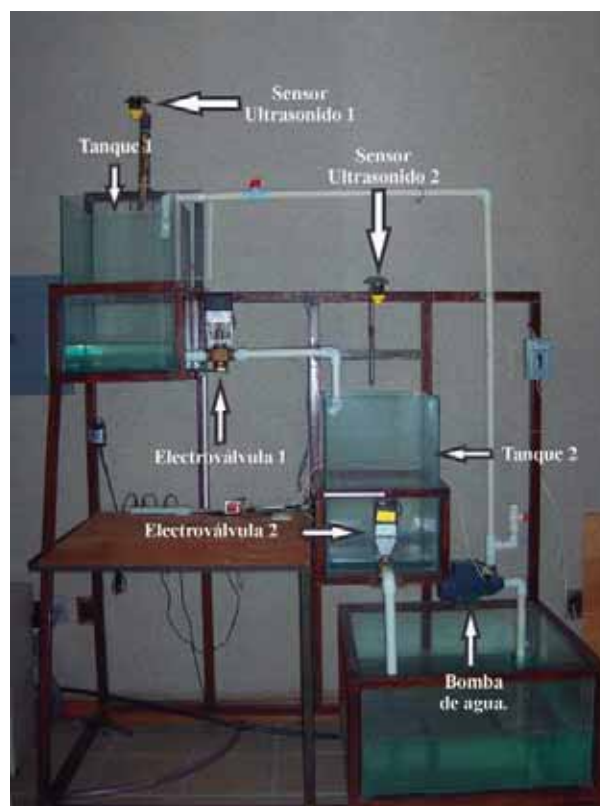


Figura 3.1: Sistema de nivel de líquido.

El sistema hidráulico construido es de 2 entradas y 2 salidas, las entradas corresponden a voltajes aplicados a las electroválvulas y las salidas corresponden a los niveles del líquido de cada tanque. Las electroválvulas trabajan con entradas de voltaje (señales de 2 volts a 10 volts) o con entradas de corriente (señales de 4mA a 20mA). En este caso se usaron las entradas del tipo de voltaje, por lo que los voltajes aplicados a las electroválvulas permiten un modificar la apertura y cierre de las mismas proporcionalmente a la entrada de voltaje aplicado [Barajas, 2010].



Figura 3.2: Tanques para almacenamiento de fluidos con capacidad de 74 litros.



Figura 3.3: Depósito de agua con una capacidad de 100 litros.

3.1.1. Sensores de ultrasonido Banner Q45U

Para determinar el nivel de líquido se utilizan sensores ultrasónicos de la marca Banner (Q45U) el cual es un detector de proximidad ultrasónico (basado en ultrasonido) como se Figura 3.4.

Las principales características de estos sensores son:

- Filtrado digital con buena inmunidad a ruido eléctrico y acústico.
- 12 a 24 volts de CD de alimentación.
- Amplio rango de trabajo en temperatura de -25° a 70°C .
- Diseño robusto para uso en ambientes duros de trabajo.
- Rango de proximidad programable.
- Distancia de sensado máxima de 1.2m.
- Distancia de sensado mínima de 20cm.
- Alimentación de voltaje de CD dentro del rango de 14 a 24 volts.



Figura 3.4: Aspecto físico de los sensores ultrasónicos Banner Q45U.

El rango de los sensores de acuerdo a las especificaciones del fabricante es de una distancia de sensado de hasta 1.2m y con una distancia lateral de 15cm, para el sistema descrito se programaron los sensores en un rango alrededor de 90 cm, esto permite sensar el nivel de líquido de los tanques desde 0cm a 60cm de altura cada uno, cuando los sensores detectan los 60 cm a la salida del sensor se tienen 10volts o 20mA y cuando el sensor detecta aproximadamente 1cm o menos la salida de voltaje del sensor es de 0v o 4mA. La Figura 3.5 muestra el rango de operación de los sensores ultrasónicos.

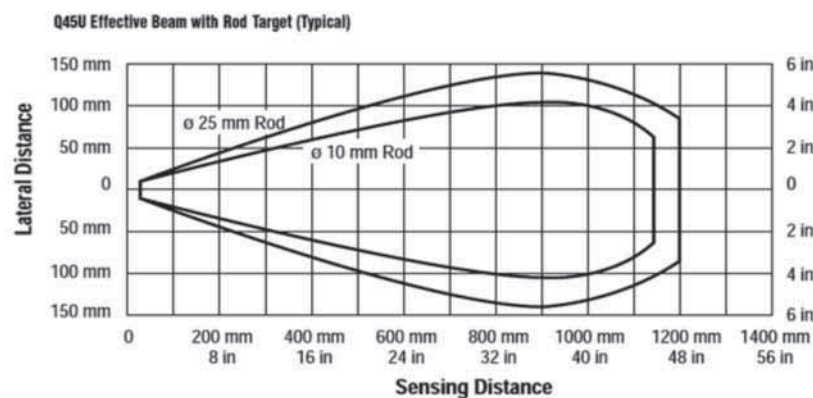


Figura 3.5: Rango de operación de los sensores ultrasónicos Banner Q45U.

3.1.1.1. Electroválvulas Honeywell, modelo ML6984

Se utilizaron electroválvulas de apertura y cierre proporcionales, (Honeywell, modelo ML6984), estas constan de un actuador y la válvula (Figura 3.6), las cuales son capaces de abrir y cerrar el flujo de agua, mediante una señal ya sea de voltaje o de corriente (la señal de voltaje va desde 2 hasta 10 volts y la señal de corriente va de 4 a 20mA). La válvula se alimenta con una fuente ya sea de 24 Vac o de 28 Vcd, para un uso más cómodo, se utilizó la señal de alimentación de CA de 24 volts.

El tiempo que tardan las válvulas motorizadas en responder desde un nivel mínimo (completamente cerrada) a un nivel máximo (completamente abierta) es de 60 segundos, con esto se obtiene que el dispositivo tarda exactamente 1 minuto en abrir y cerrar completamente, y tarda 2 segundos en recibir la señal y empezar a actuar, lo cual produce un retardo total de 1 minuto con 2 segundos.



Figura 3.6: Aspecto físico de la electroválvula con su respectivo actuador.

3.1.2. Bomba de agua

El sistema cuenta con un motor inducción monofásico Figura. 3.7 de 127 Vac de 1800 rpm con $\frac{1}{2}$ HP a 5 ampéres, que provee el caudal de entrada para los tanques.



Figura 3.7: Aspecto físico del motor de inducción.

El caudal suministrado por la bomba se obtuvo monitoreando el llenado del tanque que se encuentra directamente conectado a la bomba, se monitoreó el tiempo y la altura, el resultado se muestra en la Figura 3.8.

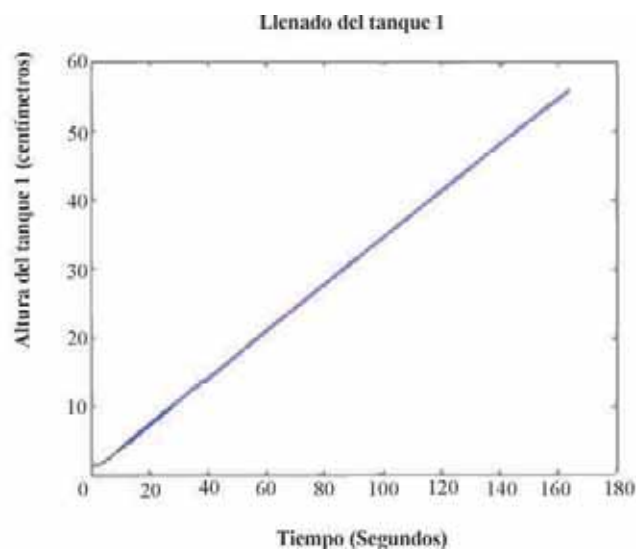


Figura 3.8: Prueba de llenado del tanque 1.

El caudal de entrada es la pendiente de la recta dada por la Figura 3.8 [Barajas, 2010] esto es:

$$Q_i = 505.45 \text{ cm}^3 / \text{seg} \quad (3.1)$$

3.2. Instrumentación del sistema de nivel de líquido

Esta sección presenta la instrumentación de los sensores de ultrasonido, electroválvulas y bomba de inducción que componen el sistema de nivel de líquido, esto para poder tener el uso de las señales de control de las electroválvulas y señales de sensado de los sensores de ultrasonido un microcontrolador.

3.2.1. Instrumentación de los sensores de ultrasonido Banner Q45U

Los sensores necesitan una alimentación de voltaje de CD dentro del rango de 14 a 24 volts y de acuerdo con la programación y al nivel detectado por los sensores, cada sensor cuenta con una salida ya sea de voltaje (de 0 a 10volts) o de corriente (de 4 a 20mA). Haciendo las pruebas de consumo de corriente pertinentes se determina que a cada uno de los sensores demandan una corriente total de 71mA cada uno en plena operación.

3.2.1.1. Etapa de alimentación del sensor de ultrasonido

Para la etapa de alimentación de los sensores se utilizó un regulador LM7824. El cual nos permite alimentar el cada sensor a 24 volts. El esquema seguido para dicho regulador y el aspecto real del regulador se muestran en la Figura 3.9 y en la Figura 3.10.

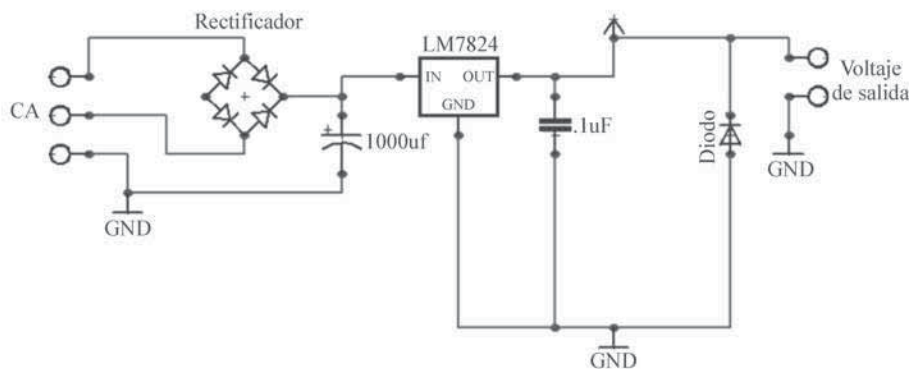


Figura 3.9: Esquemático de la fuente de alimentación de los sensores de ultrasonido.



Figura 3.10: Aspecto real de la fuente de alimentación de los sensores de ultrasonido.

3.2.1.2. Acondicionamiento de la señal del sensor de ultrasonido

La salida del sensor entrega un voltaje de variación lineal en el rango de 0 a 10 volts de CD. Para poder manejar salidas de voltaje en la red inalámbrica, se tienen que ajustar ya que se cuenta con 2 convertidores analógicos a digitales a 10 bits que manejan rangos de 0 a 5volts provenientes de un microcontrolador PIC18f4550, por lo que las salidas de los sensores se ajustan a los voltajes de 0 a 5 volts mediante un arreglo de amplificadores operacionales (con una ganancia de $\Delta=0.5$) como se observa en la Figura. 3.12, para monitorear cada una de las alturas en tiempo real (se tiene un arreglo por cada uno de los sensores de ultrasonido).

El arreglo del amplificador operacional inversor utilizado para la reducción del voltaje de 0 a 5 se muestra en la Figura. 3.11 en donde la salida de dicho arreglo está dada por la ecuación 3.2 [Boylestad, 2002].

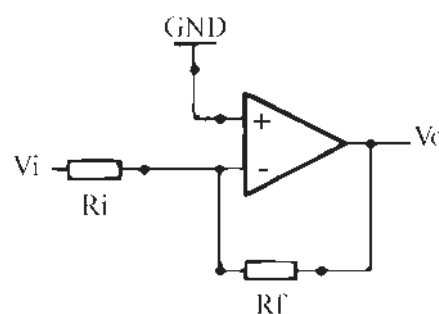


Figura 3.11: Amplificador operacional inversor.

$$V_o = -\frac{R_f}{R_i} V_i \quad (3.2)$$

La ganancia que se desea es de $\Delta=.5$ por lo que el cálculo de esta ganancia está dada por la ecuación 3.3 [Boylestad, 2002].

$$\Delta = \frac{R_f}{R_i} \quad (3.3)$$

La resistencia R_f se selecciono de 1Kohm (este valor es un valor comercial) y el cálculo de la resistencia R_i se calcula de 2Kohm (esta resistencia es una resistencia de precisión) con lo que sustituyendo en la ecuación 3.2 tenemos que la ganancia es el dado por la ecuación 3.4.

$$\Delta = \frac{R_f}{R_i} = \frac{1Kohm}{2Kohm} = .5 \quad (3.4)$$

La salida del arreglo es una señal negativa por lo que se tiene que volver a invertir la señal con el mismo arreglo mostrado en la Figura 3.10 en este caso la ganancia es de $\Delta=1$. Como es una ganancia unitaria la selección de las resistencias R_f y R_i deben ser del mismo valor en este caso se seleccionaron dos resistencias de 1Kohm como se muestra en la ecuación 3.5.

$$\Delta = \frac{R_f}{R_i} = \frac{1Kohm}{1Kohm} = 1 \quad (3.5)$$

A la salida del primer amplificador operacional inversor se tiene una señal que varía de 0 a -5 volts al colocar el segundo amplificador operacional la señal varía de 0 a 5 volts la cual es una señal adecuada para poder ser procesada directamente desde el microcontrolador y así medir el nivel de líquido de cada uno de los tanques que componen al sistema. El diagrama seguido para la construcción del arreglo de amplificadores operacionales se muestra en la Figura 3.12 y en la Figura 3.13 se muestra el aspecto real del sistema construido.

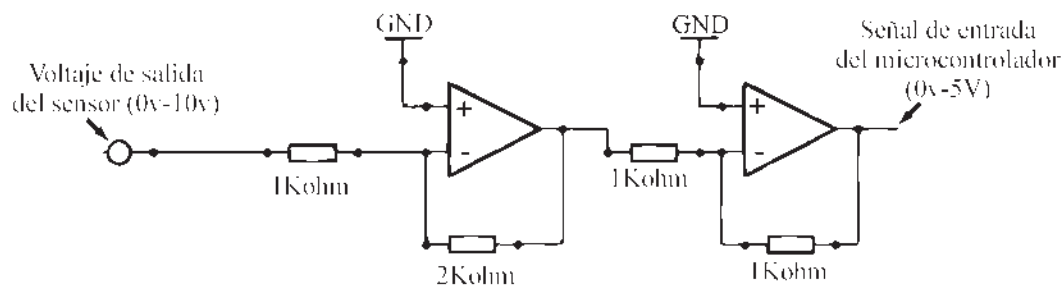


Figura 3.12: Arreglo de operacionales para reducir el voltaje de los sensores de ultrasonido.



Figura 3.13: Aspecto real del arreglo de amplificadores operacionales inversores utilizados.

3.2.2. Instrumentación de las Electroválvulas Honeywell, modelo ML6984

La entrada de voltaje es de 24 volts de corriente alterna por lo que esta etapa se implementó mediante un transformador conectado directamente a la entrada de las electroválvulas.

Las electroválvulas trabajan tanto con entradas de voltaje (señales de 2 volts a 10 volts) o con entradas de corriente (señales de 4mA a 20mA). En este caso se usaron las entradas del tipo de voltaje, por lo que los voltajes aplicados a las electroválvulas permiten un modificar la apertura y cierre de las mismas proporcionalmente a la entrada de voltaje aplicado.

Las señales de voltaje se generan mediante dos señales de modulación de ancho de pulso (PWM) de 2.9 KHz con ciclo de trabajo variable de 10 bits proveniente del microcontrolador PIC18F84550 o PIC18F82550 (una señal PWM por cada electroválvula). Como las señales PWM generadas son señales pulsantes de forma cuadrada estas contienen un conjunto de armónicos múltiplos de la frecuencia de la señal original y una componente de CD, para este caso ocupamos entregar la componente de CD y eliminar cualquier armónico presente en las señales generadas. Como la componente de CD no contiene frecuencia basta con seleccionar un filtro del tipo pasa bajas a una frecuencia menor a la de las señales PWM generadas para eliminar los armónicos. Para este caso se realizó un filtro del tipo Butterworth de segundo orden con banda de paso plana críticamente amortiguada con una frecuencia de corte de 33Hz el cual se muestra en la Figura 3.14.

3.2.2.1. Diseño del filtro pasa bajas

Existen 2 tipos de filtros, los filtros activos y los filtros pasivos.

Los filtros pasivos son implementados con dispositivos pasivos como resistencias, capacitancias e inductancias, no requieren alimentación y normalmente el voltaje de salida es menor al de entrada.

Los filtros activos son implementados con dispositivos pasivos como resistencias, capacitancias e inductancias y además de dispositivos activos, requieren alimentación y la tensión de salida puede ser amplificada [Boylestad, 2002].

Los tipos de filtros que existen son:

- LPF - Filtro pasa bajos.
- HPF - Filtro pasa altos.
- BPF – Filtro pasa banda.
- SPF – Filtro rechaza banda.
- APF – Filtro pasa todo.

Las familias de filtros activos que existen son:

- Butterworth.
- Chebyshev.
- Bessel.
- Thompson.
- Paynter.
- Elípticos.

Para este trabajo se requiere de un filtro del tipo pasa bajas ya que queremos obtener la componente de directa de la señal PWM. Por lo que se recurrió al cálculo de un filtro pasa bajas del tipo butterworth de segundo orden. Las principales características del filtro butterworth son:

- Banda de paso plana.

- Críticamente amortiguado.
- Son los más utilizados en sistemas lineales.
- Para un filtro de segundo orden $n = 2$, los parámetros α y k_f son variables constantes con los valores de $\alpha = 1.414$ y $k_f = 1$ [Boylestad, 2002].

El diagrama seguido para el armado del filtro de tipo butterworth de segundo orden es el mostrado en la Figura 3.14.

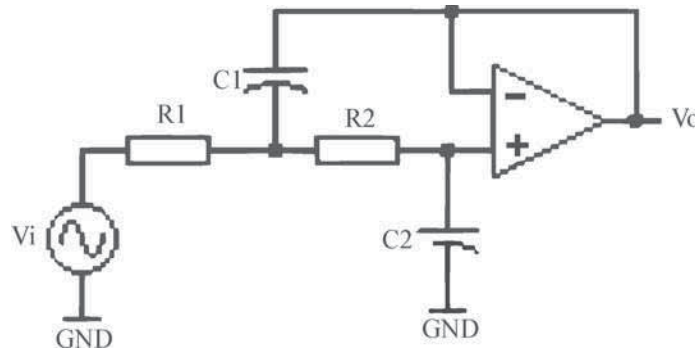


Figura 3.14: Esquema seguido para el filtro pasa bajas

Para realizar el cálculo de los componentes del filtro butterworth mostrado en la Figura 3.14 se recurre a las siguientes ecuaciones [Boylestad, 2002]:

$$R_{n1} = R_{n2} = 1ohm \quad (3.6)$$

$$C_{n1} = \frac{2}{\alpha} \quad (3.7)$$

$$C_{n2} = \frac{\alpha}{2} \quad (3.8)$$

$$k_z = \frac{R_1}{1ohm} \quad (3.9)$$

$$k_f = 2\Pi f_c \quad (3.10)$$

$$R_1 = R_{n1}k_z \quad (3.11)$$

$$R_2 = R_{n2}k_z \quad (3.12)$$

$$C_1 = \frac{C_{n1}}{k_f k_z} \quad (3.13)$$

$$C_2 = \frac{C_{n2}}{k_f k_z} \quad (3.14)$$

$$f_c = \frac{1}{2\Pi\sqrt{R_1 R_2 C_1 C_2}} \quad (3.15)$$

Se selecciona una frecuencia de corte baja en este caso se selecciono una frecuencia de $f_c = 30Hz$ y los valores de $R_1 = R_2 = 1Kohm$, $\alpha = 1.414$ y $k_f = 1$ en base a estos valores se buscan los valores de resistencia y capacitores adecuados teniendo que:

$$C_{n1} = \frac{2}{\alpha} = \frac{2}{1.414} = 1.414 \quad (3.16)$$

$$C_{n2} = \frac{\alpha}{2} = \frac{1.414}{2} = .707 \quad (3.17)$$

$$k_z = \frac{R_1}{1ohm} = \frac{1Kohm}{1ohm} = 1Kohm \quad (3.18)$$

$$k_f = 2\Pi f_c = 2 * 3.1416 * 33 = 188.49 \quad (3.19)$$

$$C_1 = \frac{C_{n1}}{k_f k_z} = \frac{1.414}{(1000)(188.4955)} = 7.5\mu F \approx 6.8\mu F \quad (3.20)$$

$$C_2 = \frac{C_{n2}}{k_f k_z} = \frac{.707}{(1000)(188.4955)} = 3.7\mu F \approx 3.3\mu F \quad (3.21)$$

$$f_c = \frac{1}{2\Pi\sqrt{R_1 R_2 C_1 C_2}} = \frac{1}{2\Pi\sqrt{(1000)(1000)(3.3\mu F)(6.8\mu F)}} \approx 33Hz \quad (3.22)$$

Una vez obtenidos los cálculos necesarios la implementación del filtro activo del tipo butterworth de segundo orden se muestra en la Figura 3.15.

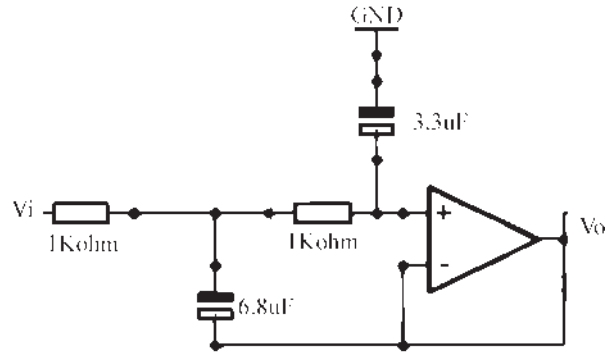


Figura 3.15: Esquema del filtro butterworth pasa bajas de segundo orden implementado.

Después de realizar el filtrado correspondiente, la señal CD estará variando de 0 a 5 volts conforme a como se modifique el ciclo de trabajo (al ciclo de trabajo del 0 % corresponde 0 volts de directa y un ciclo de trabajo de 100 % corresponden 5 volts a la salida del filtro diseñado) de cada una de las señales PWM, por lo que estas señales de CD se tienen que amplificar mediante un amplificador no inversor con una ganancia de $\Delta=2$ para ajustar las señales de 0 a 10 volts y así poder abarcar el rango completo de las electroválvulas de 2 a 10 volts, dicha implementación se muestra en la Figura 3.16.

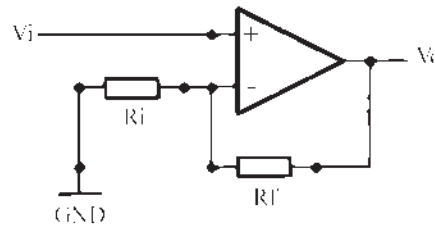


Figura 3.16: Diagrama del amplificador no inversor utilizado.

El arreglo del amplificador operacional no inversor utilizado para la elevación del voltaje de inicialmente 0-5 volts a 0-10 volts se muestra en la Figura. 3.16 en donde la salida de dicho arreglo está dada por la ecuación 3.16 [Boylestad, 2002].

$$V_o = \left(1 + \frac{R_f}{R_i}\right) V_i \quad (3.23)$$

$$\Delta = \left(1 + \frac{R_f}{R_i}\right) \quad (3.24)$$

Como es una ganancia debe de 2 la selección de las resistencias Rf y Ri deben ser del

mismo valor en este caso se seleccionaron dos resistencias de 1Kohm como se muestra en la ecuación 3.18.

$$\Delta = \left(1 + \frac{1Kohm}{1Kohm}\right) = 2 \quad (3.25)$$

El circuito diseñado para esta etapa de filtrado y amplificación con una ganancia de 2 se muestra en la Figura 3.17 (se tiene un arreglo por cada una de las electroválvulas) y el aspecto real del filtro se muestra en la Figura 3.18.

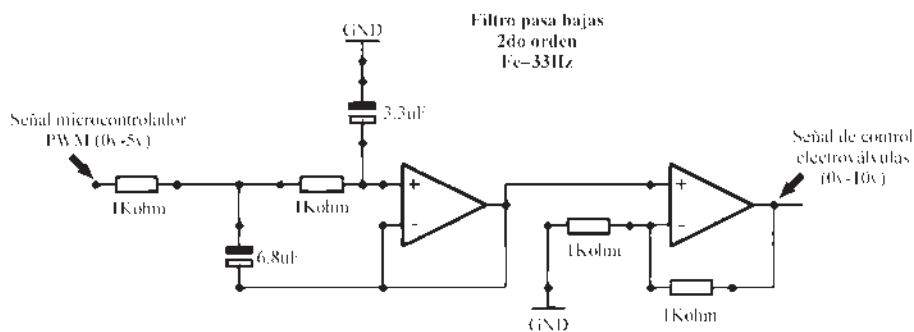


Figura 3.17: Esquema del filtro pasa bajas de segundo orden a 33Hz implementado para cada una de las señales PWM generadas.

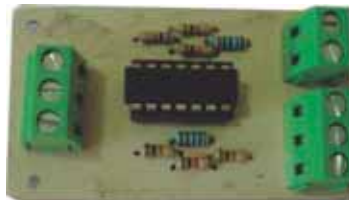


Figura 3.18: Aspecto real del filtro pasa bajas construido.

3.2.3. Etapa de potencia para el arranque de la bomba de inducción

La bomba consiste de un motor de inducción monofásico SIEMENS de 127 Vac de 1800 rpm con $\frac{1}{2}$ HP de 5 ampéres, por lo que la etapa de potencia para accionar y desactivar el motor se realizó directamente con un triac.

El disparo de tiristores y triac's supone la generación del pulso o secuencia de pulsos apropiados mediante la secuencia de circuitos analógicos y digitales aprovechando además las características de otros elementos discretos, tales como: transistores, diac's, transistores monounión, etc.

Un problema usual en el control es el de enviar señales desde un circuito de mando con una cierta tensión de referencia (microcontroladores o tarjetas de adquisición con niveles de voltaje de 5 volts) a semiconductores de potencia cuyos terminales de excitación están a distinto potencial debido a la propia naturaleza del circuito de potencia (circuitos como triac's o SCR's los cuales manejan cargas de cientos de volts). El acoplamiento óptico se realiza mediante fibras ópticas en equipos que manejan tensiones de más de 2000 V y con opto-acopladores integrados en los equipos con tensiones inferiores. El acoplamiento magnético se realiza con transformadores de impulsos de alta y baja frecuencia.

El acoplamiento óptico y el magnético tienen la ventaja de eliminar los ruidos eléctricos producidos por las corrientes de retorno en los conductores de potencia y la conmutación de dispositivos electrónicos de potencia, permitiendo realizar montajes mucho más seguros que con excitación directa.

El diagrama y aspecto real del driver construido para controlar el encendido y apagado del motor (Figura 3.19 y Figura 3.20) consta de un opto-acoplador (moc 3011), este se emplea para asilar la etapa de potencia y control y de un triac Q6010L5 para el manejo de la carga.

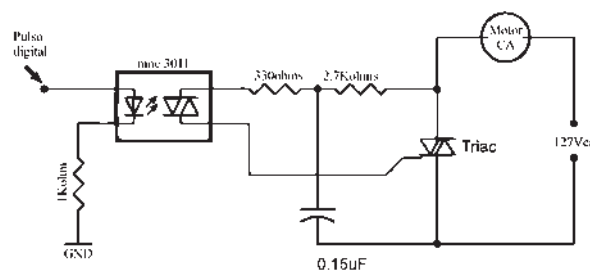


Figura 3.19: Esquemático del controlador del motor de inducción.



Figura 3.20: Aspecto real del controlador del motor de inducción.

3.3. Modelado del sistema

Para la prueba de los controladores aplicados a un sistema de nivel de líquido, se aplicó un controlador PID clásico, por medio del método de Ziegler & Nichols.

3.3.1. Método de Ziegler & Nichols

Este método consiste en aproximar el comportamiento de entrada-salida de la planta, a un sistema de primer orden con tiempo muerto, representado por la función de transferencia mostrada en la ecuación 3.26, donde: K es la ganancia estática; T es la constante de tiempo aparente; L es la constante de tiempo muerto. Las constantes de dicha ecuación se pueden determinar a partir de la respuesta al escalón unitario del proceso, si dibujamos una tangente en el punto de inflexión de la curva de reacción (respuesta del sistema ante una entrada escalón Figura 3.21), entonces la pendiente de la línea es aproximadamente: $R = K/T$; y la intersección de la tangente con el eje del tiempo nos indica el retardo de la planta (L). Los parámetros para realizar la sintonización del controlador, se obtienen a partir de las reglas mostradas en la Tabla 3 [Ogata, 1993].

$$G(s) = \frac{K}{Ts + 1} e^{-Ls} \quad (3.26)$$

Donde:

K =ganancia estática.

T =es la constante de tiempo aparente.

L =es la constante de tiempo muerto.

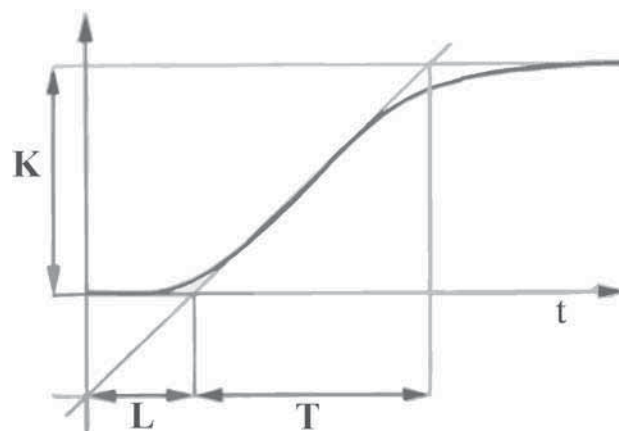


Figura 3.21: Curva S, respuesta de un proceso típico a la entrada escalón unitario.

Ziegler y Nichols sugirieron establecer los valores de K_p , T_i , y T_d de acuerdo a las fórmulas de la Tabla 3.1.

Tabla 3.1: Tabla para el cálculo de las ganancias del controlador a utilizar.

Controlador	K_p	T_i	T_d
P	$\frac{T}{L}$	∞	0
PI	$.9\frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2\frac{T}{L}$	$2L$	$\frac{L}{2}$

3.3.2. Modelado del sistema en lazo abierto

Para esta tesis se trabajó sobre un solo tanque, siendo un sistema de primer orden (el tanque 1 está conectado directamente a la bomba de agua), el modelo utilizado está basado en el modelo publicado en el artículo [Barajas, 2010], el cual es un modelo determinado para el sistema que se implementó.

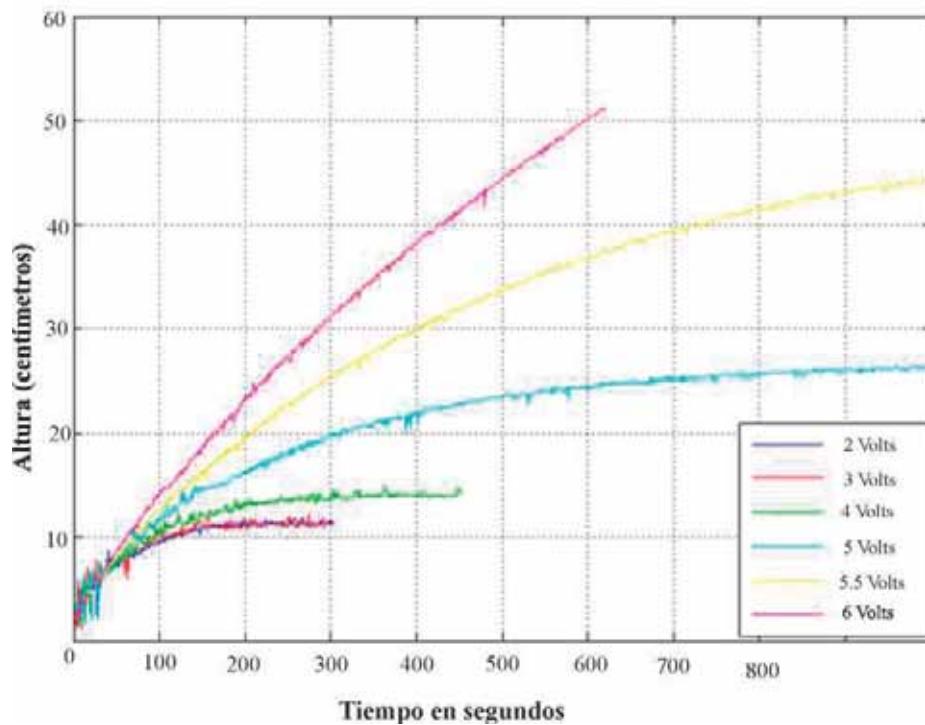


Figura 3.22: Respuestas del sistema ante entradas escalón sobre la electroválvula 1.

Para obtener el modelo del sistema y aplicar la técnica de control PID apoyados en las reglas experimentales de Ziegler & Nichols, en el artículo se procedió de manera experimental, de acuerdo al siguiente procedimiento: se propuso un punto lineal de operación que corresponde a la respuesta de una entrada del tipo escalón, en este caso se realizaron

pruebas a diferentes voltajes de entrada en las electroválvulas, es decir con aperturas diferentes (Figura 3.22). Posteriormente se busca una respuesta en lazo abierto que mejor modele al sistema, para que a la hora de hacer simulación, nuestro sistema se mantenga aproximado con respecto al sistema real, en este caso, después de hacer pruebas en simulación, se observa que la prueba a 5 volts (Aproximadamente 30 % de apertura de la electroválvula) es la que más se ajusta a la respuesta respecto a otras entradas escalón.

Ya que se obtiene la respuesta en lazo abierto, se prosigue a encontrar su función de transferencia, mediante el método de Ziegler & Nichols (Figura 3.23) [Barajas, 2010].

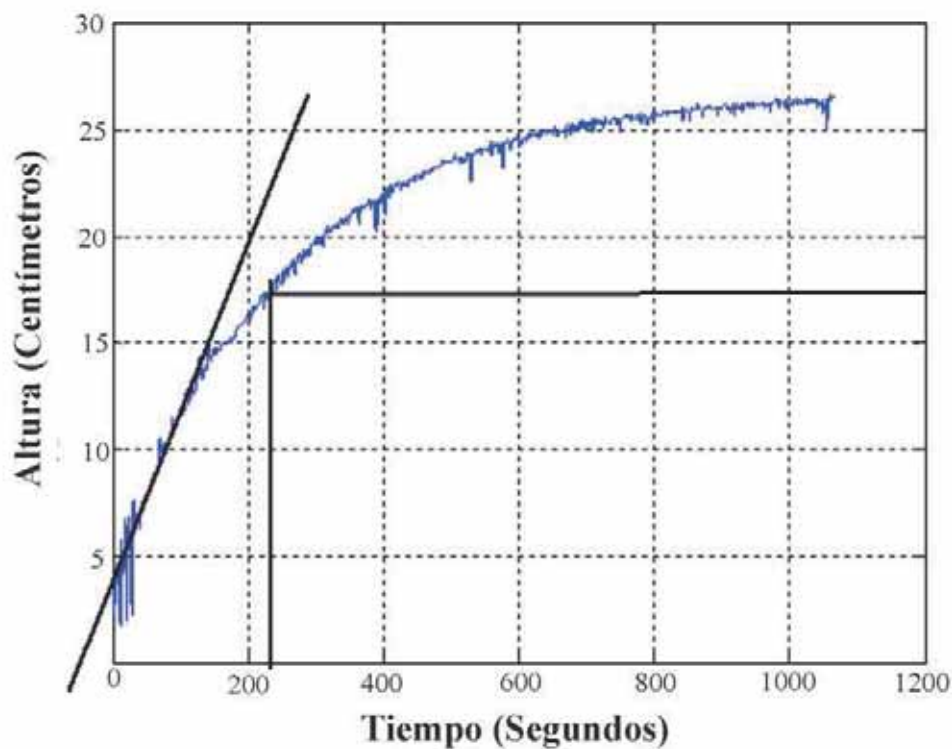


Figura 3.23: Método de Ziegler & Nichols aplicado a la prueba en lazo abierto 5V.

De la Figura 3.23 se obtienen parámetros con los que vamos a obtener la función de transferencia del modelo.

$$K = 2.4cm.$$

$$T = 216seg.$$

$$L = 10seg.$$

Así, la función de transferencia del sistema es la ecuación 3.27 [Barajas, 2010].

$$G(s) = \frac{2.4}{216s + 1} e^{-10s} \quad (3.27)$$

Ahora, con esta respuesta, se tiene un modelo aproximado que es el que se muestra en la ecuación 3.27, en la Figura 3.24, se observa la respuesta del modelo, así como la comparación con la respuesta real con varias entradas escalón.

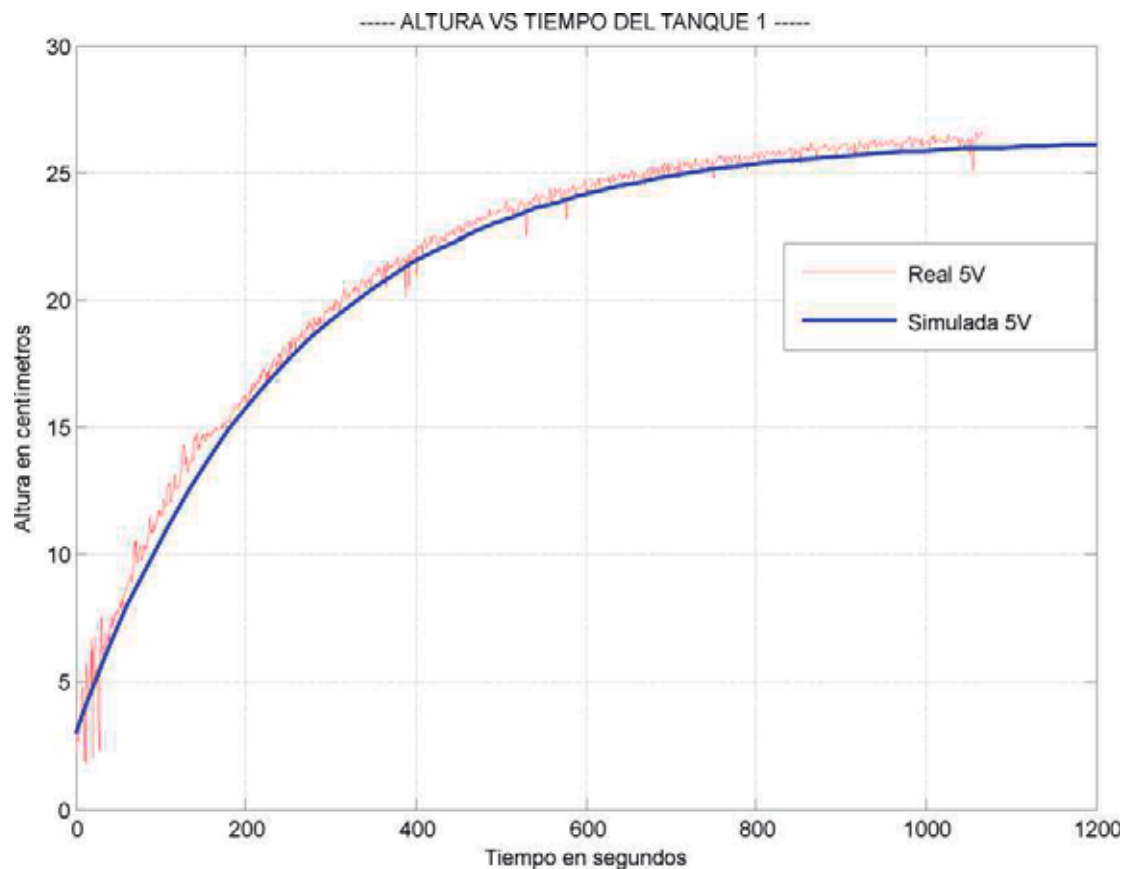


Figura 3.24: Validación del modelo obtenido.

Una vez obtenido el modelo, procedemos a calcular las ganancias del controlador PID, utilizando la Tabla 3.1 y los parámetros que ya calculamos antes teniendo así [Barajas, 2010]:

$$K_p = 1.15.$$

$$T_i = 20\text{seg.}$$

$$T_d = 5\text{seg.}$$

3.3.3. Modelado del tanque 1.

Los sensores de ultrasonido varían en un voltaje de 0 a 10 de acuerdo al nivel de líquido presente en cada uno de los tanques, si el tanque presenta un nivel mínimo 0cm la salida del sensor es de 0v y cuando se presenta un nivel máximo la salida del sensor será de 10v. El tanque 1 es el tanque superior conectado directamente a la bomba de agua. De

acuerdo a [Barajas, 2010] la ecuación que nos permite determinar la altura de este tanque en centímetros de acuerdo al voltaje presente en el sensor es:

$$Alturatanque1cm = 5.573 * voltajesensor1 + 1.45 \quad (3.28)$$

3.3.4. Modelado del tanque 2.

El tanque 2 es el tanque inferior conectado. De acuerdo a [Barajas, 2010] la ecuación que nos permite determinar la altura de este tanque en centímetros de acuerdo al voltaje presente en el sensor es:

$$Alturatanque2cm = 5.594 * voltajesensor2 + 1.964 \quad (3.29)$$

Capítulo 4

Implementación de los Esquemas de Comunicación

La comunicación inalámbrica o sin cables es aquella en la que extremos de la comunicación (emisor/receptor) no se encuentran unidos por un medio de propagación físico, sino que se utiliza la modulación de ondas electromagnéticas a través del espacio. En este sentido, los dispositivos físicos solo están presentes en los emisores y receptores de la señal, entre los cuales encontramos: antenas, computadoras portátiles, módulos de radiofrecuencia, teléfonos móviles, entre otros.

La tecnología inalámbrica utiliza ondas de radiofrecuencia de baja potencia y una banda específica, de uso libre o privada para transmitir información, entre dispositivos. Los dispositivos inalámbricos brindan flexibilidad y portabilidad mejorada para algunas aplicaciones al poder acceder en lugares donde los cables no son convenientes o viables y el objetivo es ir evitando los cables en todo tipo de comunicación.

La tendencia a la movilidad y la ubicuidad hacen que cada vez sean más utilizados los sistemas inalámbricos, actualmente el auge y la accesibilidad de las comunicaciones inalámbricas permite nuevas aplicaciones de monitoreo remoto no solo en el campo informático sino en telefonía, seguridad, domótica, procesos industriales, entre otros.

Este capítulo describe el conjunto de dispositivos diseñados y utilizados para el monitoreo remoto (empleando los esquemas de comunicación: Internet, radiofrecuencia y GSM) del sistema de nivel de líquido, así como su forma de operación.

Para el monitoreo desde Internet se utilizaron las librerías de LabVIEW® de TCP/IP y TELNET las cuales nos permiten tener acceso a otra computadora para la transmisión

de datos de manera bidireccional entre dos interfaces gráficas desde Internet para tener monitoreado y controlado el sistema de nivel de líquido de manera remota con una tarjeta basada en el PIC18F2550.

En el caso de radio frecuencia se utilizó un Xbee® para la transmisión de datos de manera bidireccional entre una interfaz gráfica y un módulo inalámbrico (diseñado en base al microcontrolador PIC18F4550 y el chip Xbee®) desde la banda libre de radio frecuencia de 2.4GHz.

Por último se propone una tarjeta basada en el módulo GSM/GPRS SM5100B y un microcontrolador PIC18F2550 para que mediante la transmisión de mensajes de texto se tenga el control manual del sistema de nivel de líquido.

4.1. Requerimientos del sistema de nivel de líquido

El sistema de nivel de líquidos que se instrumentó cuenta con 2 señales analógicas de salida una por cada sensor que varían de 0 a 10 volts de acuerdo a la altura del agua en el tanque (Figura 4.1), 2 señales analógicas de entrada una por cada electroválvula las cuales varían de 0 a 10 volts (Figura 4.2) para regular el caudal de salida de cada uno de los tanques y un pulso digital de entrada de 5 volts para accionar en encendido de la bomba de agua el cual nos da el caudal de entrada del sistema la cual se muestra en la Figura 4.3.

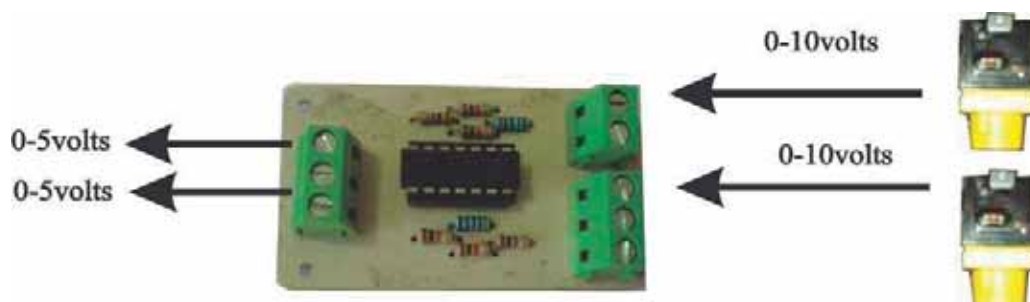


Figura 4.1: Esquema de conexión para el ajuste de las señales de los sensores de ultrasonido.

Los microcontroladores que se necesitaban seleccionar deberían cumplir con al menos la adquisición de 2 señales del tipo analógicas, la generación de al menos 2 señales PWM para su debida transformación a una señal analógica y la generación de al menos 1 señal del tipo digital, además de al menos un puerto UART o USB para la comunicación entre

la interfaz gráfica, el chip Xbee® o el módulo GSM/GPRS SM5100B.

Los microcontroladores seleccionados fueron el PIC18F4550 (Figura 4.4) y el PIC18F2550 (Figura 4.5).

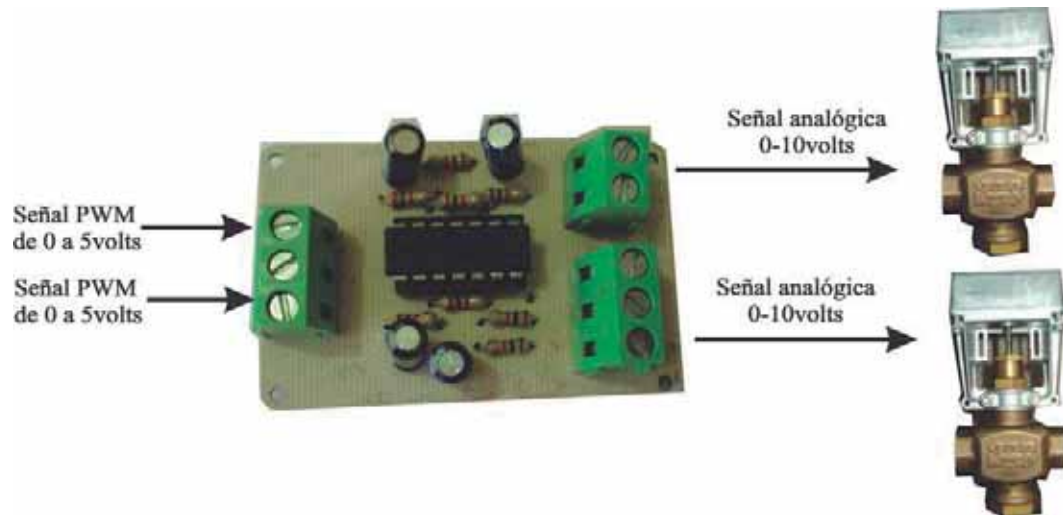


Figura 4.2: Esquema de conexión para el ajuste de las señales de las electroválvulas.



Figura 4.3: Esquema de conexión seguido para el control de la bomba de agua.

4.1.1. PIC18F4550

El microcontrolador PIC18F4550 mostrado en la Figura 4.4 cuenta con las siguientes características [Microchip, 2009]:

- Interfaz USB 2.0 de alta velocidad 12Mbit/s.
- Tipo de memoria: Flash.
- Memoria de programa: 32 Kb.
- Memoria RAM: 2048 bytes.
- EEPROM: 256 bytes.

- Voltaje de operación: 2 a 5.5 volts.
- 2 salidas PWM de salida con una resolución de 1 a 10 bits.
- Un puerto tipo USART.
- Hasta 13 canales de adquisición de señales analógico a digital con una resolución de 10 bits.
- Hasta 33 salidas digitales.

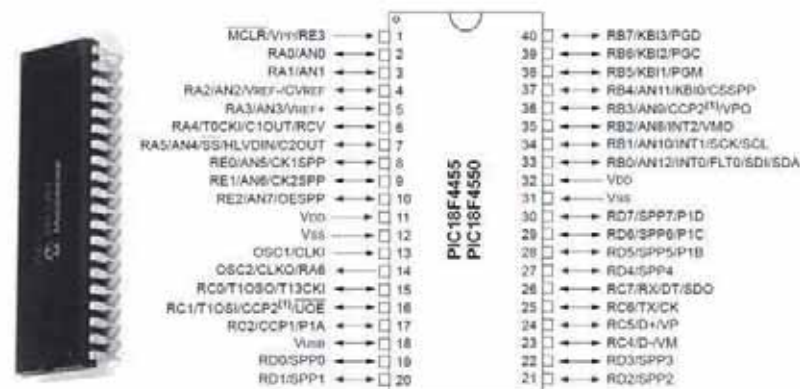


Figura 4.4: Aspecto físico y diagrama de pines del PIC18F4550.

Las características del microcontrolador lo hacen ideal para el tipo de aplicación que se realizó quizá los puertos de salida tienen una cantidad mayor a la que se ocupa en realidad, pero lo que se pensó al dejar este microcontrolador es tener pines necesarios en caso de que en proyectos futuros se requieran de controlar o medir mas sensores o actuadores.

4.1.2. PIC18F2550

El microcontrolador PIC18F2550 mostrado en la Figura 4.5 cuenta con las siguientes características [Microchip, 2009]:

- Interfaz USB 2.0 de alta velocidad 12Mbit/s.
- Tipo de memoria: Flash.
- Memoria de programa: 32 Kb.
- Memoria Ram: 2048 bytes.

- EEPROM: 256 bytes.
- Voltaje de operación: 2 a 5.5 volts.
- 2 salidas PWM de salida con una resolución de 1 a 10 bits.
- Un puerto tipo USART.
- Hasta 10 canales de adquisición de señales analógico a digital con una resolución de 10 bits.
- Hasta 22 salidas digitales.

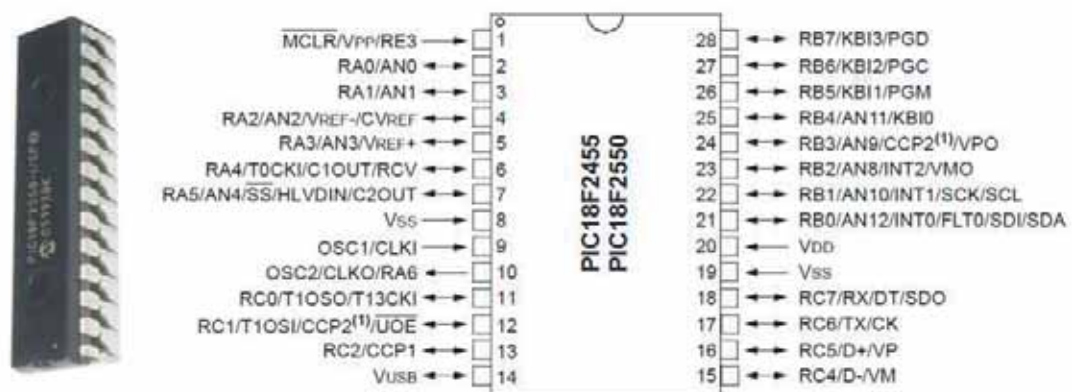


Figura 4.5: Aspecto físico y diagrama de pines del PIC18F2550.

De igual forma las características del microcontrolador lo hacen ideal para el tipo de aplicación que se realizó quizá los puertos de salida tienen una cantidad mayor a la que se ocupa en realidad, pero lo que se pensó al dejar este microcontrolador es tener pines necesarios en caso de que en proyectos futuros se requiera controlar o medir mas sensores o actuadores.

Se diseñaron 3 tarjetas inalámbricas basadas en radiofrecuencia

4.2. Tarjeta basada en el PIC18F2550

En la Figura 4.6 se muestra el diagrama de bloques de la tarjeta diseñada en base al microcontrolador PIC18F2550. La tarjeta que se diseño (Figura 4.7) está pensada para habilitar todos los pines del microcontrolador, con sus correspondientes conectores de alimentación, la habilitación del puerto USART para comunicarnos con la computadora, así como LED's indicadores que nos indiquen el funcionamiento de la tableta.

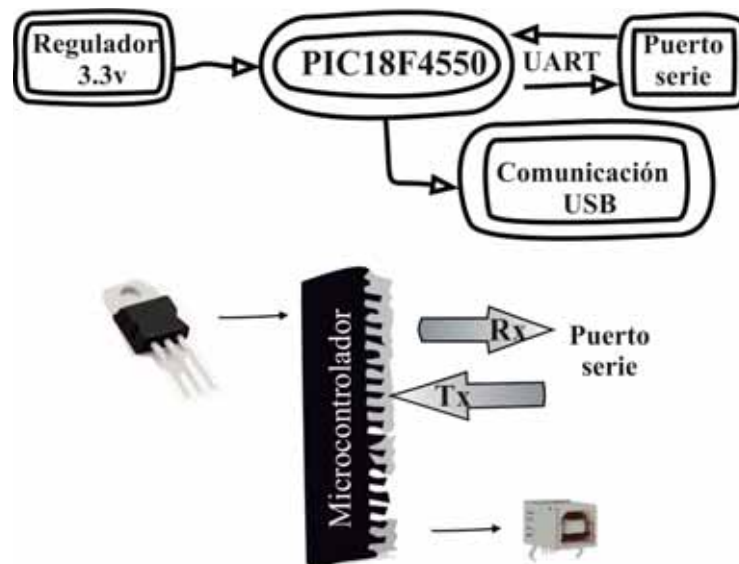


Figura 4.6: Diagrama de bloques de la tarjeta basada en el PIC18F2550.



Figura 4.7: Aspecto real de la tarjeta diseñada basada en el PIC18F2550.

En la

En la Figura 4.7 se muestra el aspecto real y se enumeran las diferentes características del circuito diseñado, las cuales son descritas a continuación.

1. La entrada de voltaje de Corriente directa se hace mediante un conector tipo jack el cual permite colocar cualquier eliminador de CD para alimentar la tarjeta.
2. El regulador que se utiliza es el LM7805 el cual por su característica el voltaje mínimo de entrada debe ser de 8 volts y permite regular una entrada de hasta 35 volts de cd para obtener a la salida 5 volts con los que se alimenta el microcontrolador, la salida máxima de corriente es de 1A a 5volts.
3. Este adaptador permite colocar algún programador externo (ICD ó PICKIT) en caso de ser necesario programar la memoria de programa del microcontrolador.

4. Es un LED el cual indica que el circuito está trabajando de manera adecuada en caso de presentarse daños ya sea en el regulador o en el microcontrolador el LED estaría apagado.
5. Este bus de salida permiten acceder a cualquier pin del microcontrolador para adquirir o transmitir señales, la corriente de salida de cada pin del microcontrolador es de hasta 20mA.
6. El conector tipo jumper permite deshabilitar la programación del microcontrolador vía USB en caso de realizar la programación por algún programador externo como el PICKIT o el ICD (programadores de MICROCHIP®).
7. Estos conectores tipo jumper permiten deshabilitar el cristal de cuarzo en caso de no ocupar la programación por el USB o en caso de necesitar operar el microcontrolador a una frecuencia específica.
8. Este conector tipo jumper permite deshabilitar la alimentación de la tarjeta vía USB, la alimentación de voltaje máximo por el puerto USB es de .5A a 5volts.
9. Este integrado es el conocido como MAX232 el cual permite adecuar las señales UART que son de 5 volts a un rango de 12 volts el cual permite conectar la tarjeta a cualquier puerto UART sin problemas de amplitud de señal a cualquier computadora.
10. Este conector permite programar el microcontrolador vía USB o bien transmitir datos desde el microcontrolador a la computadora.
11. El conector DB9 es un conector universal para la comunicación USART, por lo que la tarjeta se puede conectar a cualquier computadora.
12. Este push button permite reiniciar el microcontrolador en caso de ser necesario.
13. Este push button permite junto con el push botton número 12 programar el microcontrolador vía USB en caso de ser necesario.

La forma de conexión que se siguió se muestra en la Figura 4.10 en donde se observa el esquema general seguido para el control de las electroválvulas, adquisición de las señales

de los sensores de ultrasonido, la señal de control de la bomba de agua y la transmisión de datos vía UART.

En la Figura 4.8 se puede observar que el microcontrolador es el encargado de adquirir las señales provenientes de la instrumentación de los sensores de ultrasonido para medir el nivel de líquido de cada uno de los tanques del sistema mediante los canales de conversión analógico a digital 0 y 1, este mismo se encarga de generar las señales PWM mediante los 2 canales PWM con los que cuenta, el pulso digital mediante el pin RB7 y la transmisión de los datos vía USART. Esto nos permite tener el control y monitoreo de todas las variables del sistema desde la interfaz gráfica la que se muestra en el capítulo 5.

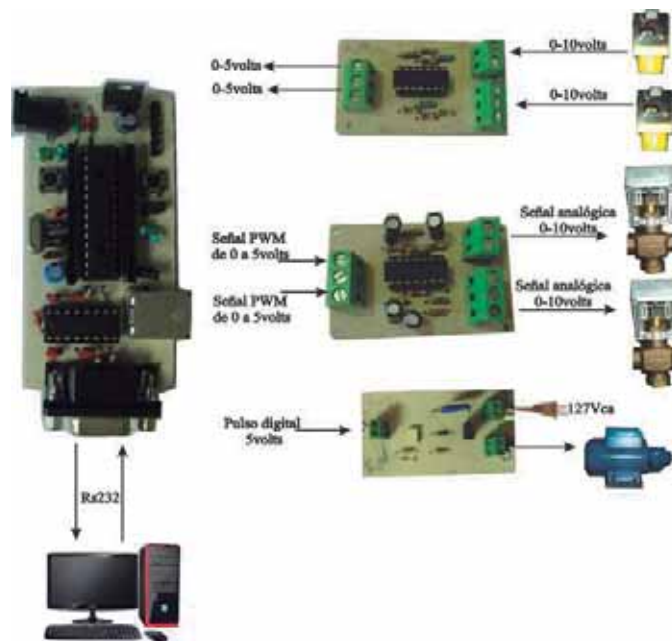


Figura 4.8: Esquema general de conexión del sistema mediante la tarjeta basada en el PIC18F2550.

4.3. Implementación del protocolo ZigBee

Durante el capítulo 2 se describió el funcionamiento general de los módulos de radio frecuencia Xbee®. En esta sección se presenta el diseño de los 3 módulos realizados basados en el microcontrolador PIC18F4550, el microcontrolador PIC18F2550 y el puerto RS232, los cuales permiten realizar el monitoreo y control del sistema de nivel de líquido de manera remota.

4.3.1. Chip de radiofrecuencia Xbee®

Los módulos Xbee® permiten controlarlos mediante dispositivos (Figura. 4.9 y Figura. 4.10) que manejen directamente un puerto de comunicación serial asíncrona. A través de este puerto, el módulo Xbee® puede comunicarse con cualquier dispositivo o puerto con comunicación UART (por ejemplo algún microcontrolador o algún puerto RS232 de una computadora).

Los dispositivos que cuentan con una interfaz UART pueden ser conectados directamente al módulo Xbee® como se muestra en la Figura 4.9 y la Figura 4.10.



Figura 4.9: Esquema general de conexión entre una computadora y un microcontrolador mediante módulos Xbee®.



Figura 4.10: Esquema de comunicación entre 2 microcontroladores.

Los datos que llegan al módulo Xbee® a través del pin 3 como una señal asíncrona serial. Cada byte de datos se compone de un bit de inicio, 8 bits de datos iniciando con el bit menos significativo y un bit de paro.

La comunicación serial dependerá de que los 2 puertos UART (el del microcontrolador y el del módulo de radio frecuencia) sean configurados con opciones compatibles (la velocidad de transmisión, la paridad, el bit de inicio, el bit de paro, y los bits de datos).

Para esta tesis los módulos que se utilizan, se operan en un modo transparente. Cuando se opera los módulos en esta forma, los datos recibidos actúan como un cable RS232 inalámbrico. En este modo todo lo que ingresa por el pin 3 (DI Data in), es guardado en el buffer de entrada y luego transmitido y todo lo que ingresa como paquete RF, es guardado en el buffer de salida y luego enviado por el pin 2 (DO Data out).

Cuando los datos entran en el módulo de RF a través del pin DI (pin 3), los datos se almacenan en el buffer DI hasta que puedan ser procesados. Cuando los datos de radio frecuencia son recibidos, los datos entran en el buffer de DO y son enviados del puerto serie al dispositivo que está conectado al módulo Xbee®. Una vez que el buffer se satura (la capacidad del buffer es de 100bytes), los datos adicionales de entrada de RF se pierden [Manual Xbee®, 2007].

4.3.1.1. Comandos AT

Para modificar o leer los parámetros del módulo Xbee®, el módulo debe entrar en modo de comando (un estado en el los caracteres de entrada son interpretados como comandos de instrucciones). Existen dos modos de comando para configurar los módulos: el modo de comando AT y el modo de comando API. Los módulos se programaron mediante el modo de comandos AT [Manual Xbee®, 2007].

Para entrar al modo de comandos AT se necesitan enviar una secuencia de 3 caracteres los cuales son “+++” y esperar un tiempo determinado por el comando AT GT (por default este valor es de 0x3E8 lo que es 1 segundo aproximadamente) antes y después de introducir la secuencia de caracteres, en general se tiene que cuidar la siguiente secuencia:

- No enviar caracteres al módulo por un segundo.
- Introducir en el módulo los caracteres “+++” en un segundo.
- No enviar caracteres al módulo por un segundo.

Para la transmisión de comandos AT al módulo se sigue la sintaxis propuesta en la Figura 4.11.

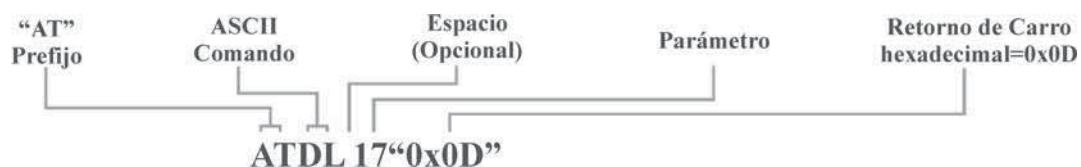


Figura 4.11: Sintaxis para enviar comandos AT al módulo Xbee®.

Para la leer comandos AT al módulo se sigue la sintaxis propuesta en la Figura 4.12.



Figura 4.12: Sintaxis para leer comandos AT al módulo.

Para modificar valores de los comandos AT, los cambios deben ser guardados en la memoria no volátil con el comando AT WR (escritura de comandos). De lo contrario, los parámetros se restablecen a los valores previamente guardados después de que el módulo se resetea.

Cuando se envía un comando al módulo, el módulo analizará y ejecutará el comando. Tras la ejecución con éxito de un comando, el módulo devuelve un mensaje "OK". Si la ejecución de un comando de resultados en un error, el módulo devuelve un "Error".

Para salir del modo de comandos AT simplemente se envía el comando ATCN seguido de un retorno de carro (hexadecimal 0x0D). Si no son recibidos comandos AT válidos dentro de un tiempo específico determinado por el comando CT, el módulo saldrá del modo de comandos AT [Manual Xbee®], 2007].

La Tabla 4.1 muestra los comandos AT especiales.

Tabla 4.1: Lista de comandos AT especiales para el módulo Xbee®.

Comando.	Nombre y descripción.	Parámetro.
WR	Write. Escribe los cambios en la memoria no volátil.	-
RE	Restore Defaults. Restaura el módulo a los parámetros de fábrica.	-
FR	Software Reset. Responde inmediatamente con un "OK" y resetea el módulo.	-

La Tabla 4.2 muestra los comandos AT que permiten guardar las opciones seleccionadas. La Tabla 4.3 muestra los comandos AT encargados de la interfaz serial.

Tabla 4.2: Lista de comandos AT que permiten guardar las opciones seleccionadas para el módulo Xbee®.

Comando.	Nombre y descripción.	Parámetro.
CT	Tiempo de espera del modo de comandos AT. Permite leer o escribir el período de inactividad del módulo Xbee® (sin comando válidos recibidos).	2 - 0xFFFF [x 100 ms]
CN	Salida del modo de comandos AT. Permite salir del modo de comandos.	-
GT	Tiempo de espera. Permite cambiar el tiempo de silencio antes y después de entrar en el modo de comandos AT.	2 - 0x0CE4 [x 1 ms]

Tabla 4.3: Lista de comandos AT encargados de la interfaz serial para el módulo Xbee®.

Comando.	Nombre y descripción.	Parámetro.
BD	Velocidad de datos. Permite leer o escribir la velocidad de transmisión de datos del módulo Xbee®.	0 - 7 (estándar) 0 = 1200 bps 1 = 2400 2 = 4800 3 = 9600 4 = 19200 5 = 38400 6 = 57600 7 = 115200
RO	Tiempo de espera. Permite leer o escribir el tiempo de espera antes de la transmisión.	0 - 0xFF
AP	Habilitación del modo API. Permite habilitar o deshabilitar el modo API.	0 =deshabilitado. 1 =habilitado.
NB	Paridad. Permite leer o escribir las opciones de paridad.	0 = 8 bit sin paridad. 1 = 8 bit incluso. 2 = 8 bit impare. 3 = 8 bit marca. 4 = 8 bit espacio.
DB	Potencia de la señal. Permite leer la potencia de la señal en dB del último paquete recibido. Se presenta el valor absoluto por ejemplo: 0x58 = -88 dBm.	

La Tabla 4.4 muestra los comandos AT encargados de la red y de la seguridad de la red.

Tabla 4.4: Lista de comandos AT encargados de la red y de la seguridad de la red para el módulo Xbee®.

Comando.	Nombre y descripción.	Parámetro.
CH	Channel. Permite leer o colocar el número del canal usado para transmitir y recibir datos entre los módulos a utilizar.	(Xbee®) 0x0B - 0x1A (Xbee-PRO®) 0x0C - 0x17
ID	PAN ID. Permite leer o escribir la PAN (Área de red personal), es el número de red sobre el cual van a trabajar los dispositivos.	0 - 0xFFFF
DH	Dirección de destino parte alta. Lee o escribe la parte alta es decir 32 bits de los 64 de la dirección de destino. Este comando solo se usa junto con el comando DL para transmitir mediante un direccionamiento de 64 bits de otra forma este registro permanece en 0x00000000 y el valor de DL debe ser menor a 0xFFFF.	0 - 0xFFFFFFFF
DL	Dirección de destino parte baja. Lee o escribe la parte baja de la dirección de destino. Este comando solo se usa junto con el comando DH para transmitir mediante un direccionamiento de 64 bits de otra forma este registro debe ser menor a 0xFFFF y el valor de DH debe ser 0x00000000.	0 - 0xFFFFFFFF
MY	Dirección del dispositivo. Permite leer o escribir la dirección de 16 bits del módulo a usar dentro de la PAN.	0 - 0xFFFF
SH	Número serial parte alta. Permite leer la parte alta de 32 bits del dispositivo.	0 - 0xFFFFFFFF [read-only]
SL	Número serial parte baja. Permite leer la parte baja de 32 bits del dispositivo.	0 - 0xFFFFFFFF [read-only]
ND	Nodo Descubierta. Descubre y reporta todos los módulos encontrados dentro de la PAN, envía la información de los dispositivos encontrados de la siguiente manera: MY<CR> SH<CR> SL<CR> DB<CR> NI<CR><CR>	
CE	Coordinador habilitado. Permite colocar o leer las opciones del coordinador.	0 - 1 0 = End Device 1 = Coordinator

4.3.1.2. Modo de operación punto a punto

Los módulos Xbee® fueron configurados para operar en una topología de red punto a punto y por lo tanto no depender de una relación maestro-esclavo. Esto significa que los módulos permanecen sincronizados sin el uso de configuraciones de algún módulo Xbee® que funcione como maestro. Esta configuración tiene características como rápidos tiempos de sincronización y rápidos tiempos de arranque en frío y se adapta a una amplia gama de aplicaciones.

Cada paquete de datos que es enviado por un módulo contiene la dirección del dispositivo que está mandando los datos y una dirección destino. El módulo conforme a las especificaciones del protocolo IEEE 802.15.4 soporta direcciones cortas de 16 bits o largas de 64 bits. Una dirección de 64 bits es asignada de fábrica esta dirección no puede ser alterada de ninguna manera. Las direcciones cortas de 16 bits pueden ser configuradas manualmente [Manual Xbee®, 2007].

4.3.1.3. Direcciones cortas de 16 bits

Como ya se mencionó los módulos pueden ser configurados usando direcciones de destino y direcciones propias de 16 bits usando el comando AT adecuado. Para la configuración de la dirección personal del módulo se utiliza el comando AT MY ($MY < 0xFFFFE$). Colocando el comando AT DH como $DH=0$ permitirá configurar la dirección destino como una dirección corta de 16 bits. Para comunicar 2 módulos usando direcciones cortas.

Para comunicar dos módulos a partir de direcciones cortas de 16 bits es necesario la dirección destino del módulo transmisor debe ser la dirección propia (comando AT MY) del módulo receptor y de igual forma para el receptor la dirección destino debe ser la dirección asignada al módulo transmisor.

En la Tabla 4.5 y en la Figura 4.13 se observa la configuración de 2 módulos para obtener un direccionamiento de 16bit.

Tabla 4.5: Ejemplo de configuración de los módulos Xbee® usando un direccionamiento de 16bits.

Comando AT	Módulo Xbee 1	Módulo Xbee 2
My(Dirección propia)	0x01	0x02
DH(Dirección destino parte alta)	0	0
DL(Dirección destino parte baja)	0x02	0x01



Figura 4.13: Direccionamiento de 16bits de dos módulos Xbee®.

Los valores de la Tabla 4.5 son valores en hexadecimal.

Este tipo de direccionamiento se pueden crear una red de módulos Xbee® de hasta 65535 dispositivos [Manual Xbee®, 2007].

4.3.1.4. Direcciones largas de 64 bits

Todos los módulos Xbee® contienen un número serial (connotado con los comandos AT SH y SL) los cuales son usados para direccionamientos con direcciones de 64bits. Cuando el comando AT MY esta deshabilitado, es decir cuando el comando AT MY se encuentra con los valores MY=0xFFFF o con el valor MY=0xFFFFE, las direcciones de los módulos son colocadas con la dirección IEEE de 64 bits, a esta dirección se puede acceder mediante los comandos AT SL y SH [Manual Xbee®, 2007].

4.3.1.5. Configuración de los módulos Xbee®

Para configurar los módulos que se van a utilizar hay que tener en cuenta que cualquier módulo que deseemos comunicar con otro deben de trabajar en el mismo canal (comando AT CH), deben de operar en la misma red de área personal PAN ID (comando AT ID) y deben tener definidas las direcciones destino (comandos AT DH y comando AT DL) ya que esta será el modo de comunicarse con algún módulo en específico.

Teniendo en cuenta lo anterior la topología que se siguió fue una configuración punto a punto con un direccionamiento de 16 bits como se muestra en la Figura 4.16.

En la Figura 4.14 se observa la forma en la cual se conecta el sistema. El monitoreo se realiza con una interfaz gráfica desde una computadora conectada a una tarjeta inalámbrica basada en un módulo Xbee® que contiene una interfaz RS232 (sección 4.3.5), esta tableta transmite a la computadora los datos que llegan desde el módulo Xbee® los cuales son procesados por la computadora para mostrar los datos, al mismo tiempo este módulo RS232 transmite los datos de control del sistema al módulo Xbee® USB basada

en el microcontrolador PIC18F4550 (sección 4.3.3 y sección 4.3.4) el cual esta conectado directamente al sistema y tiene el control directo de todos los dispositivos que conforman al sistema de nivel de líquido [Manual Xbee®, 2007].

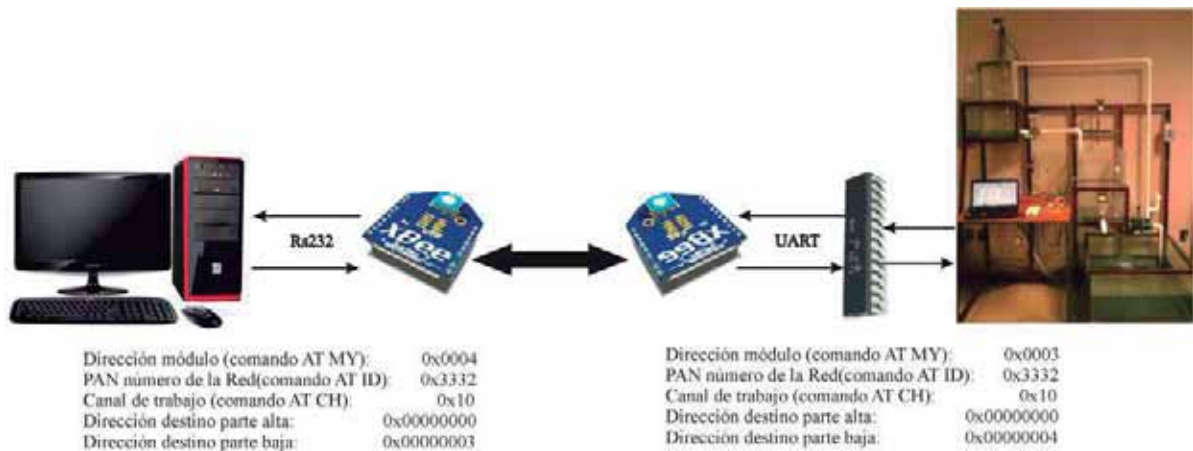


Figura 4.14: Configuración punto a punto utilizado para los módulos Xbee®.

4.3.2. Tarjeta adaptadora del módulo Xbee®

En la Figura 4.15 y Figura 4.16 se muestran las dimensiones del Xbee-PRO® y Xbee®, donde se puede observar que la separación de los pines es de 2mm, esta separación entre pines no es compatible con el estándar de separación entre pines (como el de las protoboards o bases de pines) el cual es de .25mm.



Figura 4.15: Dimensiones del módulo XbeePRO®.

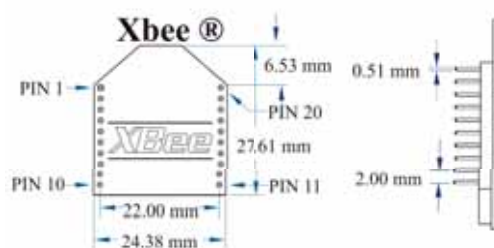


Figura 4.16: Dimensiones del módulo Xbee®.

Se diseñó y construyó una placa para el módulo Xbee®. Esta placa conecta los 20 pines de la Xbee® al estándar de dos hileras paralelas de pines con un espaciamiento de .25mm. En la Figura 4.17 y la Figura 4.18 se muestran los adaptadores construidos para los módulos. La Figura 4.18 muestra el esquemático diseñado. Y en la Figura 4.19 muestra el aspecto real de la tarjeta construida.



Figura 4.17: Esquemático del adaptador para el módulo Xbee®.

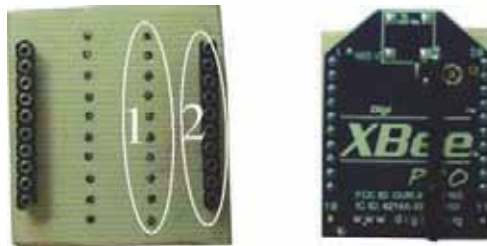


Figura 4.18: Aspecto real del adaptador para el módulo Xbee®.

En la Figura 4.18 se puede observar que la marca colocada con el número 1 muestra el conector utilizado con una separación de 2mm para colocar el módulo Xbee® y la marca con el número 2 muestra la tira de pines colocada la cual contiene la separación estándar entre pines de .25mm. Entre las 2 tiras de pines con separación de .25mm que se observan en el centro el espacio entre las dos tiras de pines es de .1cm haciendo posible conectar el Xbee® a una protoboard sin problemas.

4.3.3. Tarjeta inalámbrica basada en el PIC18F4550

En la Figura 4.19 se muestra el diagrama de bloques de las tarjetas inalámbricas diseñada en base al microcontrolador PIC18F2550 y el PIC18F4550.

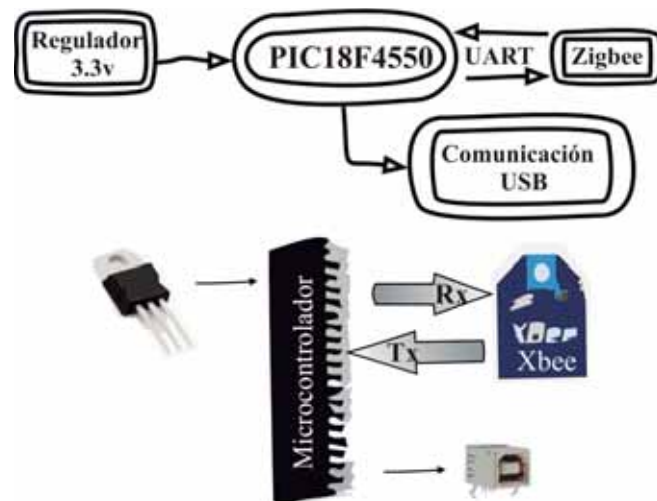


Figura 4.19: Diagrama de bloques de las tarjetas inalámbricas basadas en el módulo Xbee®.

La primer tarjeta construida para manejar el módulo Xbee® se diseño en base al microcontrolador PIC18F4550 (Figura. 4.20), la cual está pensada para habilitar todos los pines del microcontrolador y el módulo Xbee®, con sus correspondientes conectores y reguladores de alimentación (tanto para el microcontrolador el cual trabaja a un voltaje desde 2.2 a 5volts, como para el Xbee® el cual trabaja a un voltaje de 3.3 volts), la habilitación del puerto USART para comunicarnos directamente con el módulo Xbee®, la habilitación de LED's indicadores que nos indiquen el funcionamiento de la tableta y la habilitación del puerto USB con el que cuenta el microcontrolador.



Figura 4.20: Aspecto real de la tarjeta inalámbrica basada en el PIC18F4550.

En la tarjeta el microcontrolador PIC18F4550 opera a 5 volts y el módulo Xbee® opera a 3.3 volts para poder conectar el puerto USART del microcontrolador y el puerto UART del módulo Xbee® es necesario hacer un ajuste a las señales, en el caso del pin receptor del microcontrolador la señal puede ser recibida directamente del pin transmisor

del módulo ya que el microcontrolador opera con señales desde 2.5 a 5 volts, para el caso del pin transmisor del microcontrolador (señal de 5volts) esta debe ser reducida a una señal de 3.3 volts ya que la entrada del módulo Xbee® máxima es de 3.3volts. El arreglo utilizado para la reducción de la señal de transmisión del microcontrolador al módulo Xbee® se muestra en la Figura 4.21.

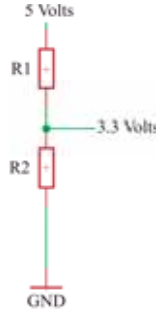


Figura 4.21: Acoplo de la señal de 5 a 3.3 volts.

En la Figura 4.21 se muestra el acoplo que se realiza para poder conectar directamente el microcontrolador con el módulo Xbee®, el acoplo se realiza mediante un arreglo resistivo en base a dos resistencias R_1 y R_2 .

Para realizar el cálculo de R_1 y R_2 se tiene que $V_i = 5volts$ y $V_o = 3.3volts$ proponiendo un valor comercial para R_1 de $R_1 = 1.7Kohms$ se recurre a la ecuación 4.4 para el cálculo de la resistencia R_2 .

$$V_i = i * R_1 + i * R_2 \quad (4.1)$$

$$i = \frac{V_i}{R_1 + R_2} \quad (4.2)$$

$$V_o = i * R_2 \quad (4.3)$$

$$R_2 = \frac{V_o * R_1}{V_i - V_o} \quad (4.4)$$

$$R_2 = \frac{3.3Volts * 1.7Kohms}{5Volts - 3.3Volts} = 3.3Kohms \quad (4.5)$$

En la Figura 4.20 se enumeran las diferentes características del circuito diseñado, las cuales son descritas a continuación:

1. La entrada de voltaje de Corriente directa se hace mediante un conector tipo jack el cual permite colocar cualquier eliminador o regulador de cd para alimentar la tarjeta.

2. Este bus de salida permite acceder a cualquier pin del módulo Xbee® para adquirir o transmitir señales.
3. El regulador que se utiliza para alimentar el microcontroladores el LM7805 el cual por su característica el voltaje mínimo de entrada debe ser de 8 volts y permite regular una entrada de hasta 35 volts de cd para obtener a la salida 5 volts, la salida máxima de corriente es de 1A a 5v.
4. El regulador para alimentar el módulo Xbee® es el LD33V el cual permite obtener a la salida 3.3 volts, la salida máxima de corriente es de .5A a 3.3v.
5. Cuando el módulo Xbee® está en operación este indicador tipo LED se encuentra en constante parpadeo (prendido y apagado) lo cual indica que el módulo está operando de manera adecuada, en caso de que el modulo esté conectado y este indicador no prenda quiere decir que el módulo Xbee® utilizado esta dañado.
6. Cuando el módulo Xbee® está en operación y algún tipo de paquete con información llega al módulo este indicador tipo LED indica en cuanto a la intensidad de la luz que presenta la potencia de la señal que llega al módulo Xbee®.
7. Cuando el módulo Xbee® está en operación este indicador tipo LED indica en cuando algún tipo de paquetes con información son recibidos de otro módulo.
8. Cuando el módulo Xbee® está en operación este indicador tipo LED indica en cuando algún tipo de paquetes con información son transmitidos a otro módulo.
9. Es un indicador tipo LED el cual indica que el circuito está trabajando de manera adecuada en caso de presentarse daños ya sea en el regulador o en el microcontrolador el LED estaría apagado.
10. Esta base de pines nos permite conectar la tarjeta adaptadora del chip Xbee® (Figura. 4.18) descrita en la sección 4.3.2.
11. Este bus de salida permiten acceder a cualquier pin del módulo Xbee® para adquirir o transmitir señales.
12. Estos buses de salida permiten acceder a la tierra (GND), el cual podría ser utilizado para conectar otros dispositivos o sensores.

13. Este conector tipo jumper permite deshabilitar la programación del microcontrolador vía USB en caso de realizar la programación por algún programador externo como el PICKIT o el ICD (programadores de MICROCHIP®).
14. Este conector tipo jumper permite deshabilitar la alimentación del de la tarjeta vía USB, la alimentación de voltaje máximo por el puerto USB es de .5A a 5volts.
15. Este conector tipo jumper permite deshabilitar la transmisión de paquetes de información del microcontrolador al módulo Xbee®.
16. Este conector tipo jumper permite deshabilitar la recepción de paquetes de información del microcontrolador al módulo Xbee®.
17. Este bus de salida nos permiten acceder a los 3.3volts del regulador LD33V, los cuales podrían ser utilizados para alimentar otro tipo de circuito, la salida máxima de corriente es de .5A a 3.3volts.
18. Este conector nos permite programar el microcontrolador vía USB o bien transmitir datos desde el microcontrolador a la computadora.
19. Este bus de salida permiten acceder a los 5volts del regulador LM7805, los cuales podrían ser utilizados para alimentar otro tipo de circuito, la salida máxima de corriente es de 1A a 5volts.
20. Este push button permite junto con el push button del reset (componente 21) programar el microcontrolador vía USB en caso de ser necesario.
21. Este push button permite reiniciar el microcontrolador en caso de ser necesario.
22. Este push button permite reiniciar el módulo Xbee® en caso de ser necesario.
23. Este bus de salida permiten acceder al puerto B del microcontrolador.
24. Este bus de salida permiten acceder al puerto D del microcontrolador.
25. Este adaptador permite colocar algún programador externo (ICD ó PICKIT) en caso de ser necesario reprogramar la memoria de programa del microcontrolador
26. Este circuito es el microcontrolador que puede ser remplazado en caso de estar dañado ya que cuenta con una base de pines adecuada.

27. Este bus de salida permiten acceder al puerto A del microcontrolador.
28. Este bus de salida permiten acceder al puerto E del microcontrolador.
29. Este bus de salida permiten acceder al puerto C del microcontrolador.

En la Figura 4.22 se muestra el aspecto real de la tarjeta inalámbrica diseñada junto con a tarjeta adaptadora del módulo Xbee® (Figura 4.18).

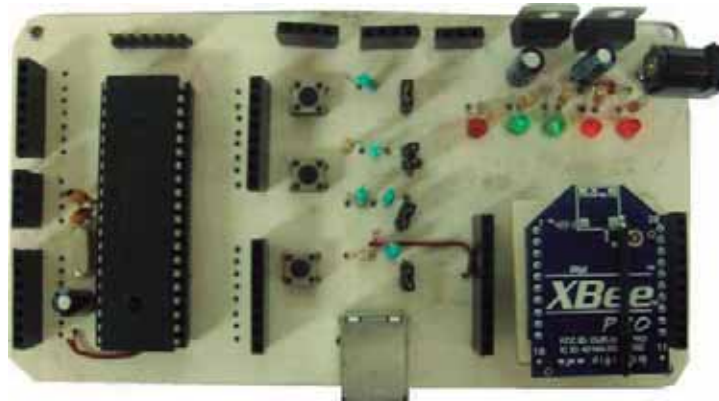


Figura 4.22: Aspecto real de la tarjeta inalámbrica basada en el PIC18F4550 con la tarjeta adaptadora del módulo Xbee®.

La forma de conexión que se siguió se muestra en la Figura 4.23, el esquema general seguido para el control de las electroválvulas, adquisición de las señales de los sensores de ultrasonido, la señal de control de la bomba de agua y la transmisión de datos vía UART.

En la Figura 4.23 se puede observar que el microcontrolador es el encargado de adquirir las señales provenientes de la instrumentación de los sensores de ultrasonido para medir el nivel de líquido de cada uno de los tanques del sistema mediante los canales de conversión analógico a digital 0 y 1, este mismo se encarga de generar las señales PWM mediante los 2 canales PWM con los que cuenta, el pulso digital mediante el pin RB7 y la transmisión de los datos vía USART. Esto nos permite tener el control y monitoreo de todas las variables del sistema desde la interfaz gráfica de la que se hablara en el capítulo 5.

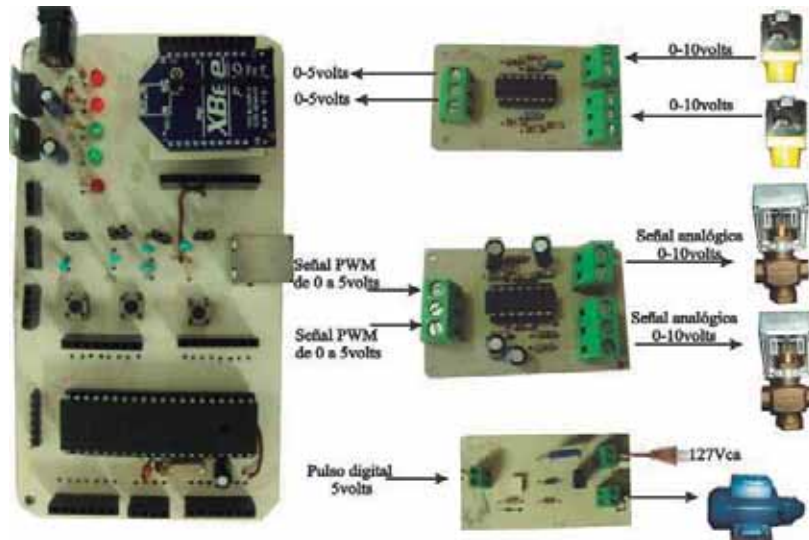


Figura 4.23: Esquema general de conexión del sistema mediante la tarjeta inalámbrica basada en el PIC18F4550.

4.3.4. Tarjeta inalámbrica basada en el PIC18F2550

La segunda tarjeta construida para manejar el módulo Xbee® se diseñó en base al microcontrolador PIC18F2550 (Figura. 4.24), la cual está pensada para habilitar todos los pines del microcontrolador y el módulo Xbee®, con sus correspondientes conectores y reguladores de alimentación (tanto para el microcontrolador el cual trabaja a un voltaje desde 2.2 a 5 volts, como para el Xbee® el cual trabaja a un voltaje de 3.3 volts), la habilitación del puerto USART para comunicarnos directamente con el módulo Xbee®, la habilitación de LED's indicadores que nos indiquen el funcionamiento de la tableta y la habilitación del puerto USB con el que cuenta el microcontrolador.

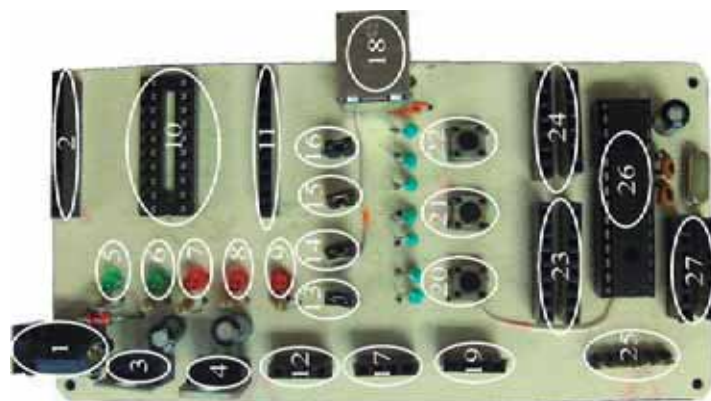


Figura 4.24: Aspecto real de la tarjeta inalámbrica basada en el PIC18F2550.

En la Figura 4.24 se muestra el aspecto real de la tarjeta diseñada. En la tarjeta el

microcontrolador PIC18F2550 opera a 5 volts y el módulo Xbee® opera a 3.3 volts para poder conectar el puerto USART del microcontrolador y el puerto UART del módulo Xbee® es necesario hacer un ajuste a las señales, en el caso del pin receptor del microcontrolador la señal puede ser recibida directamente del pin transmisor del módulo ya que el microcontrolador opera con señales desde 2.5 a 5 volts, para el caso del pin transmisor del microcontrolador (señal de 5volts) esta debe ser reducida a una señal de 3.3 volts ya que la entrada del módulo Xbee® máxima es de 3.3volts.

En la Figura 4.21 se muestra el acoplo que se hace para poder conectar directamente el microcontrolador con el módulo Xbee®, el acoplo se realiza mediante un arreglo resistivo en base a dos resistencias una de 1.7Kohms y otra de 3.3Kohms, cualquier señal de 5 volts a la salida del arreglo será de 3.3 volts.

En la Figura 4.24 se enumeran las diferentes características del circuito diseñado, las cuales son descritas a continuación:

1. La entrada de voltaje de Corriente directa se hace mediante un conector tipo jack el cual permite colocar cualquier eliminador de CD para alimentar la tarjeta.
2. Este bus de salida permite acceder a cualquier pin del módulo Xbee® para adquirir o transmitir señales.
3. El regulador que se utiliza para alimentar el microcontroladores el LM7805 el cual por su característica el voltaje mínimo de entrada debe ser de 8 volts y permite regular una entrada de hasta 35 volts de cd para obtener a la salida 5 volts, la salida máxima de corriente es de 1A a 5v.
4. El regulador para alimentar el módulo Xbee® es el LD33V el cual permite obtener a la salida 3.3 volts, la salida máxima de corriente es de .5A a 3.3v.
5. Cuando el módulo Xbee® está en operación este indicador tipo LED se encuentra en constante parpadeo (prendido y apagado) lo cual indica que el módulo está operando de manera adecuada, en caso de que el módulo esté conectado y este indicador no prenda quiere decir que el módulo Xbee® utilizado esta dañado.
6. Cuando el módulo Xbee® está en operación y algún tipo de paquete con información llega al módulo este indicador tipo LED indica en cuanto a la intensidad de la luz que presenta la potencia de la señal que llega al módulo Xbee®.

7. Cuando el módulo Xbee® está en operación este indicador tipo LED indica en cuando algún tipo de paquetes con información son recibidos de otro módulo.
8. Cuando el módulo Xbee® está en operación este indicador tipo LED indica en cuando algún tipo de paquetes con información son transmitidos a otro módulo.
9. Es un indicador tipo LED el cual indica que el circuito está trabajando de manera adecuada en caso de presentarse daños ya sea en el regulador o en el microcontrolador el LED estaría apagado.
10. Esta base de pines permite conectar la tarjeta adaptadora del chip Xbee® (Figura. 4.18) descrita en la sección 4.3.2.
11. Este bus de salida permiten acceder a cualquier pin del módulo Xbee® para adquirir o transmitir señales.
12. Estos buses de salida permiten acceder a la tierra (GND), el cual podría ser utilizado para conectar otros dispositivos o sensores.
13. Este conector tipo jumper permite deshabilitar la programación del microcontrolador vía USB en caso de realizar la programación por algún programador externo como el PICKIT o el ICD (programadores de MICROCHIP®).
14. Este conector tipo jumper permite deshabilitar la alimentación del de la tarjeta vía USB, la alimentación de voltaje máximo por el puerto USB es de .5A a 5volts.
15. Este conector tipo jumper permite deshabilitar la transmisión de paquetes de información del microcontrolador al módulo Xbee®.
16. Este conector tipo jumper permite deshabilitar la recepción de paquetes de información del microcontrolador al módulo Xbee®.
17. Este bus de salida nos permiten acceder a los 3.3volts del regulador LD33V, los cuales podrían ser utilizados para alimentar otro tipo de circuito, la salida máxima de corriente es de .5A a 3.3volts.
18. Este conector permite programar el microcontrolador vía USB o bien transmitir datos desde el microcontrolador a la computadora.

19. Este bus de salida permiten acceder a los 5volts del regulador LM7805, los cuales podrían ser utilizados para alimentar otro tipo de circuito, la salida máxima de corriente es de 1A a 5volts.
20. Este push button permite junto con el push button del reset (componente 21) programar el microcontrolador vía USB en caso de ser necesario.
21. Este push button permite reiniciar el microcontrolador en caso de ser necesario.
22. Este push button permite reiniciar el módulo Xbee® en caso de ser necesario.
23. Este bus de salida permiten acceder al puerto B del microcontrolador.
24. Este bus de salida permiten acceder al puerto D del microcontrolador.
25. Este adaptador permite colocar algún programador externo (ICD ó PICKIT) en caso de ser necesario reprogramar la memoria de programa del microcontrolador
26. Este circuito es el microcontrolador que puede ser remplazado en caso de estar dañado ya que cuenta con una base de pines adecuada.
27. Este bus de salida permiten acceder al puerto A del microcontrolador.
28. Este bus de salida permiten acceder al puerto E del microcontrolador.
29. Este bus de salida permiten acceder al puerto C del microcontrolador.

En la Figura 4.25 se muestra el aspecto real de la tarjeta inalámbrica diseñada junto con a tarjeta adaptadora del módulo Xbee® (Figura 4.18).



Figura 4.25: Aspecto real de la tarjeta inalámbrica basada en el PIC18F2550 con la tarjeta adaptadoora del módulo Xbee®.

La forma de conexión que se siguió se muestra en la Figura 4.32 se muestra el esquema general seguido para el control de las electroválvulas, adquisición de las señales de los

sensores de ultrasonido, la señal de control de la bomba de agua y la transmisión de datos vía UART.

En la Figura 4.26 se puede observar que el microcontrolador es el encargado de adquirir las señales provenientes de la instrumentación de los sensores de ultrasonido para medir el nivel de líquido de cada uno de los tanques del sistema mediante los canales de conversión analógico a digital 0 y 1, este mismo se encarga de generar las señales PWM mediante los 2 canales PWM con los que cuenta, el pulso digital mediante el pin RB7 y la transmisión de los datos vía USART. Esto nos permite tener el control y monitoreo de todas las variables del sistema desde la interfaz gráfica de la que se hablara en el capítulo 5.

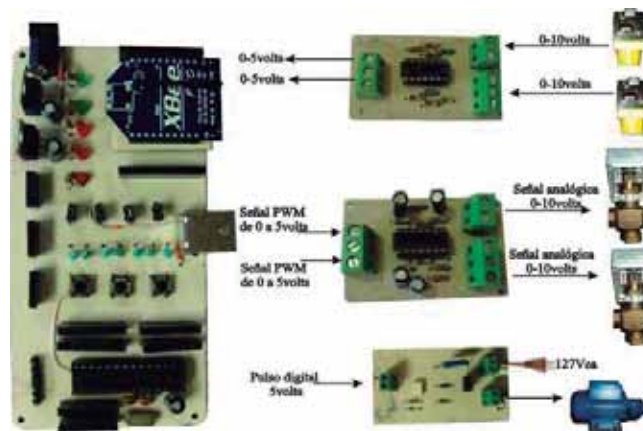


Figura 4.26: Esquema general de conexión del sistema mediante la tarjeta inalámbrica basada en el PIC18F2550.

4.3.5. Tarjeta inalámbrica basada en el puerto RS232

En la Figura 4.27 se muestra el diagrama de bloques de la tarjeta inalámbrica diseñada en base al puerto RS232.

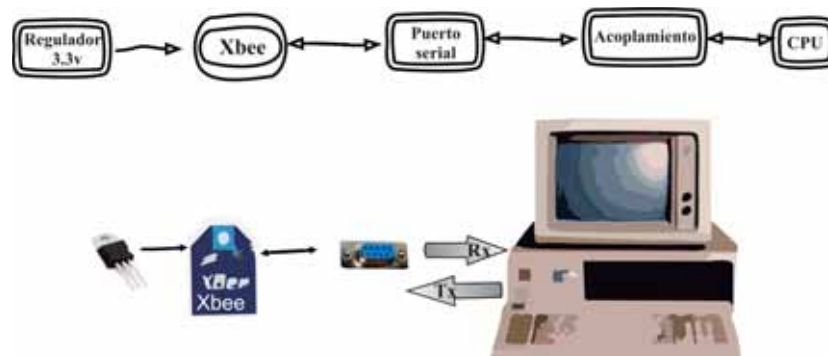


Figura 4.27: Diagrama de bloques de las tarjeta inalámbrica basada en el puerto RS232.

La tercer tarjeta construida para manejar el módulo Xbee® se diseño en base a

la necesidad de construir un dispositivo capaz de transmitir datos, así como adquirir y mostrar en la computadora los datos de cualquier tarjeta construida, por lo que se diseñó una tarjeta con comunicación RS232 (Figura. 4.28), los circuitos principales de esta tarjeta son el módulo Xbee® y el circuito integrado max232 el cual permite habilitar la comunicación RS232 para comunicarnos con la computadora. La tarjeta está pensada para habilitar todos los pines del módulo Xbee®, con sus correspondientes conectores y reguladores de alimentación (para el Xbee® el cual trabaja a un voltaje de 3.3 volts y para el max232 que trabaja a 5volts), la habilitación del puerto USART para comunicarnos directamente con el módulo Xbee® y la habilitación de LED's indicadores que indiquen el funcionamiento de la tableta.

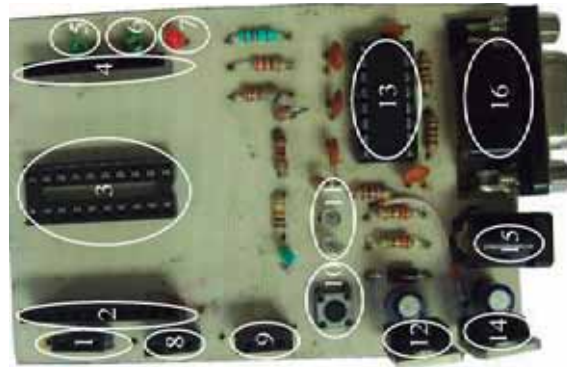


Figura 4.28: Aspecto real de la tarjeta inalámbrica basada la comunicación RS232.

En la Figura 4.28 se muestra el aspecto real de la tarjeta diseñada. El circuito max232 opera a 5 volts y el módulo Xbee® opera a 3.3volts para poder conectar el puerto USART del max232 y el puerto UART del módulo Xbee® es necesario hacer un ajuste a las señales, en el caso del pin receptor del max232 la señal puede ser recibida directamente del pin transmisor del módulo ya que el max232 opera con señales desde 2.5 a 5 volts, para el caso del pin transmisor del max232 (señal de 5volts) esta debe ser reducida a una señal de 3.3 volts ya que la entrada del módulo Xbee® máxima es de 3.3volts. En la Figura 4.21 se muestra el acoplo que se hace para poder conectar directamente el max232 con el módulo Xbee®, el acoplo se realiza mediante un arreglo resistivo en base a dos resistencias una de 1.7Kohms y otra de 3.3Kohms, cualquier señal de 5 volts a la salida del arreglo será de 3.3 volts.

En la Figura 4.28 se enumeran las diferentes características del circuito diseñado, las cuales son descritas a continuación:

1. Estos buses de salida permiten acceder a la tierra (GND), el cual podría ser utilizado

para conectar otros dispositivos o sensores.

2. Este bus de salida permiten acceder a cualquier pin del módulo Xbee® para adquirir o transmitir señales.
3. Esta base de pines permite conectar la tarjeta adaptadora del chip Xbee® (Figura. 4.18) descrita en la sección 4.3.2.
4. Este bus de salida permiten acceder a cualquier pin del módulo Xbee® para adquirir o transmitir señales.
5. Cuando el módulo Xbee® está en operación este indicador tipo LED indica cuando algún otro módulo se comunica con éxito módulo Xbee®.
6. Cuando el módulo Xbee® está en operación y algún tipo de paquete con información llega al módulo este indicador tipo LED, nos indica en cuanto a la intensidad de la luz que presenta la potencia de la señal que llega al módulo Xbee®.
7. Cuando el módulo Xbee® está en operación este indicador tipo LED se encuentra en constante parpadeo (prendido y apagado) lo cual indica que el módulo está operando de manera adecuada, en caso de que el módulo esté conectado y este indicador no prenda quiere decir que el módulo Xbee® utilizado esta dañado.
8. Este bus de salida permite acceder a los 3.3volts del regulador LD33V, los cuales podrían ser utilizados para alimentar otro tipo de circuito, la salida máxima de corriente es de .5A a 3.3volts.
9. Este bus de salida permite acceder a los 5volts del regulador LM7805, los cuales podrían ser utilizados para alimentar otro tipo de circuito, la salida máxima de corriente es de 1A a 5volts.
10. Este push button permite reiniciar el módulo Xbee® en caso de ser necesario.
11. Estos indicadores tipo LED indican que el circuito está trabajando de manera adecuada en caso de presentarse daños ya sea algún regulador o el módulo Xbee® cualquiera de estos 2 LED's estarían apagados.
12. El regulador que se utiliza para alimentar el circuito integrado max232 es el LM7805 el cual por su característica el voltaje mínimo de entrada debe ser de 8 volts y permite

regular una entrada de hasta 35 volts de cd para obtener a la salida 5 volts, la salida máxima de corriente es de 1A a 5v.

13. Este integrado es el conocido como MAX232 el cual permite adecuar las señales UART que son de 5 volts a un rango de 12 volts el cual permite conectar la tarjeta a cualquier puerto UART sin problemas de amplitud de señal a cualquier computadora.
14. El regulador para alimentar el módulo Xbee® es el LD33V el cual permite obtener a la salida 3.3 volts, la salida máxima de corriente es de .5A a 3.3v.
15. La entrada de voltaje de Corriente directa se hace mediante un conector tipo jack el cual permite colocar cualquier eliminador o regulador de cd para alimentar la tarjeta.
16. El conector DB9 es un conector universal para la comunicación USART, por lo que la tarjeta se puede conectar a cualquier computadora.

En la Figura 4.29 se muestra el aspecto real de la tarjeta inalámbrica RS232 diseñada junto con la tarjeta adaptadora del módulo Xbee® (Figura 4.18).

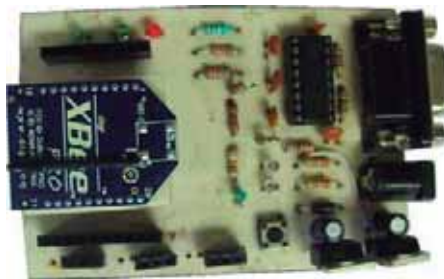


Figura 4.29: Aspecto real de la tarjeta inalámbrica RS232 con la tarjeta adaptadoora del módulo Xbee®.

La forma de conexión que se siguió se muestra en la Figura 4.38 se muestra el esquema general seguido para el control de las electroválvulas, adquisición de las señales de los sensores de ultrasonido, la señal de control de la bomba de agua y la transmisión de datos vía UART.

En la Figura. 4.30 se puede observar que el microcontrolador es el encargado de adquirir las señales provenientes de la tarjeta inalámbrica conectada al sistema de nivel de líquido, este modulo está conectado a una interfaz gráfica la cual permite visualizar y controlar

todos los dispositivos del sistema. Esto nos permite tener el control y monitoreo de todas las variables del sistema desde la interfaz gráfica de la que se hablara en el capítulo 5.

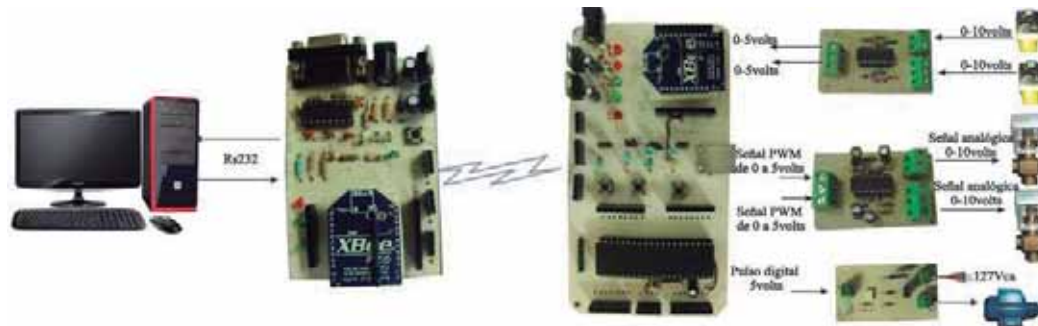


Figura 4.30: Esquema general de conexión del sistema mediante la tarjeta inalámbrica RS232.

4.3.6. Funciones programadas

Una vez que se tienen las tarjetas inalámbricas basadas en los microcontroladores PIC18f4550 y PIC18F2550 así como la tarjeta RS232, es necesario que los microcontroladores tengan el control de los módulos Xbee® es decir que puedan, configurar los módulos que se van a utilizar modificando el canal de transmisión (comando AT CH), deben de ser capaces de modificar la red de área personal PAN ID (comando AT ID) y deben tener definidas y poder modificar las direcciones destino (comandos AT DH y comando AT DL) ya que estas funciones serán el modo de comunicarse con algún módulo en específico [Manual Xbee®, 2007].

Las funciones que se programaron se basan en tener en cuenta las características anteriores es decir se tiene una función para modificar la dirección propia del módulo mediante la función `atmy()`, las dirección destino con la que se determina con que módulo se va a comunicar mediante la función `atdl()`, el canal de frecuencia en el que se va a comunicar `atch()`, la red de área local mediante la función `atid()`, una función para entrar en modo comandos AT para configurar el módulo mediante la función `in_at()`, una función para guardar los cambios `atwr()` y una función para salir de línea de comandos, así como funciones de transmisión y recepción de datos vía RS232 mediante las interrupciones con las que cuentan los microcontroladores utilizados en las tarjetas [Manual Xbee®, 2007].

En la Figura 4.31 se observa los bytes necesarios para entrar en línea de comandos AT, de color azul (bytes: "2B 2B 2B") se tienen los bytes que el microcontrolador debe transmitir al módulo Xbee® para entrar en línea de comandos AT, de color rojo (bytes

“4F 4B 0D”) son los bytes que el microcontrolador necesita esperar son los bytes que indican que se esta en modo de línea de comandos AT para configurar el módulo Xbee®. En términos de programación del microcontrolador la forma de mandar llamar el modo de comandos AT seria con la función `in_at();`.

```
+++OK      2B 2B 2B 4F 4B 0D
           in_at();
```

Figura 4.31: Bytes necesarios para entrar en línea de comandos AT del módulo Xbee®.

En la Figura 4.32 se observa los bytes necesarios para configurar las direcciones, de color azul se tienen los bytes que el microcontrolador debe transmitir al módulo Xbee® para modificar la dirección interna del modulo, los primeros 4 bytes (bytes: “61 74 6D 79”) es una trama que se tiene que cuidar siempre ya que es la que le indica al módulo que se va a modificar la dirección interna del módulo en la red, los siguientes 4 bytes (byte1, byte2, byte3, byte4) son los correspondientes al valor de la dirección que se va a usar en este caso para trabajar en un direccionamiento de 16 bits este valor no puede ser mayor a FFFE (`atmy < FFFE`) el byte 1 es el mas significativo y byte 4 es el menos significativo, el último byte de azul (byte:0D) es el que le indica al módulo que la trama de bytes que transmite el microcontrolador acabo, de color rojo (bytes “4F 4B 0D”) son los bytes que el microcontrolador necesita esperar, indican que se introdujo de manera correcta la dirección en el módulo Xbee®. En términos de programación del microcontrolador la forma de modificar la dirección del módulo seria con la función `atmy(byte1,byte2,byte3,byte4);`.

```
atmy0002   61 74 6D 79 30 30 30 32 0D
OK         4F 4B 0D
           atmy(byte1,byte2,byte3,byte4);
```

Figura 4.32: Bytes necesarios para entrar configurar la dirección interna del módulo Xbee®.

En la Figura 4.33 se observa los bytes necesarios configurar el canal de frecuencia a utilizar, de color azul se tienen los bytes que el microcontrolador debe transmitir al módulo Xbee® para modificar el canal del modulo, los primeros 4 bytes (bytes: “61 74 63 68”) es una trama que se tiene que cuidar siempre ya que es la que le indica al módulo que se va a modificar el canal del módulo en la red, los siguientes 2 bytes (byte1, byte2) son los correspondientes al valor del canal que se va a utilizar el byte1 es el más significativo y

byte 2 es el menos significativo se puede recurrir a la Tabla 2.5 para seleccionar el canal a usar, el último byte de azul (byte:0D) es el que le indica al módulo que la trama de bytes que transmite el microcontrolador acaba, de color rojo (bytes “4F 4B 0D”) son los bytes que el microcontrolador necesita esperar, indican que se introdujo de manera correcta el canal en el módulo Xbee®. En términos de programación del microcontrolador la forma de modificar la dirección del módulo sería con la función `atch(byte1,byte2);`.

```
atch15      61 74 63 68 31 35 0D
OK          4F 4B 0D

atch(byte1,byte2);
```

Figura 4.33: Bytes necesarios para modificar el canal de transmisión del módulo Xbee®.

En la Figura 4.34 se observa los bytes necesarios para configurar el número de la red de área local en la que se va a trabajar, de color azul se tienen los bytes que el microcontrolador debe transmitir al módulo Xbee® para modificar el número de red, los primeros 4 bytes (bytes: “61 74 69 64”) es una trama que se tiene que cuidar siempre ya que es la que le indica al módulo que se va a modificar el número de la red y los siguientes 4 bytes (byte1, byte2, byte3, byte4) son los correspondientes al valor de la red que se va a utilizar el byte1 es el más significativo y byte 4 es el menos significativo, el último byte de azul (byte:0D) es el que le indica al módulo que la trama de bytes que transmite el microcontrolador acaba, de color rojo (bytes “4F 4B 0D”) son los bytes que el microcontrolador necesita esperar, indican que se introdujo de manera correcta el canal en el módulo Xbee®. En términos de programación del microcontrolador la forma de modificar la dirección del módulo sería con la función `atid(byte1,byte2,byte3,byte4);`.

```
atid0005    61 74 69 64 30 30 30 35 0D
OK          4F 4B 0D

atid(byte1,byte2,byte3,byte4);
```

Figura 4.34: Bytes necesarios para modificar el número de red del módulo Xbee®.

En la Figura 4.35 se observa los bytes necesarios para introducir la dirección del módulo con el que se va a comunicar, de color azul se tienen los bytes que el microcontrolador debe transmitir al módulo Xbee® para modificar el número de red, los primeros 4 bytes (bytes: “61 74 64 6C”) es una trama que se tiene que cuidar siempre ya que es la que le indica al módulo que se va a modificar el número de la red y los siguientes 8 bytes

(byte1, byte2, byte3, byte4, byte5, byte6, byte7, byte8) son los correspondientes al valor de la dirección del módulo con el que se quiere comunicar, el byte1 es el más significativo y byte 8 es el menos significativo, el último byte de azul (byte:0D) es el que le indica al módulo que la trama de bytes que transmite el microcontrolador acabo, de color rojo (bytes “4F 4B 0D”) son los bytes que el microcontrolador necesita esperar, indican que se introdujo de manera correcta el la dirección en el módulo Xbee®. En términos de programación del microcontrolador la forma de modificar la dirección del módulo sería con la función `atdl(byte1,byte2,byte3,byte4,byte5,byte6,byte7,byte8);`.



The image shows a serial monitor interface. On the left, the text 'atdl00000000' is displayed in blue, with 'OK' in red below it. On the right, a series of bytes are shown: '61 74 64 6C 30 30 30 30 30 30 30 30 0D' in blue, and '4F 4B 0D' in red below them.

```
atdl(byte1,byte2,byte3,byte4,byte5,byte6,byte7,byte8);
```

Figura 4.35: Bytes necesarios para modificar el número de red del módulo Xbee®.

En la Figura 4.36 se observa los bytes necesarios para guardar los cambios en el módulo, de color azul se tienen los bytes que el microcontrolador debe transmitir al módulo Xbee® (bytes: “61 74 77 72 0D”) es una trama que se tiene que cuidar siempre ya que es la que le indican al módulo que guarde los cambios, el último byte de azul (byte:0D) es el que le indica al módulo que la trama de bytes que transmite el microcontrolador acabo de color rojo (bytes “4F 4B 0D”) son los bytes que el microcontrolador necesita esperar, indican que se guardaron de manera correcta los cambios en el módulo Xbee®. En términos de programación del microcontrolador la forma de guardar los cambios sería con la función `atwr();`.



The image shows a serial monitor interface. On the left, the text 'atwr' is displayed in blue, with 'OK' in red below it. On the right, a series of bytes are shown: '61 74 77 72 0D' in blue, and '4F 4B 0D' in red below them.

```
atwr();
```

Figura 4.36: Bytes necesarios para modificar el número de red del módulo Xbee®.

En la Figura 4.37 se observa los bytes necesarios salir de la línea de comandos AT, de color azul se tienen los bytes que el microcontrolador debe transmitir al módulo Xbee® (bytes: “61 74 63 6E 0D”) es una trama que se tiene que cuidar siempre ya que es la que le indican que salga de línea de comandos AT, el último byte de azul (byte:0D) es el que le indica al módulo que la trama de bytes que transmite el microcontrolador acabo, de color

rojo (bytes “4F 4B 0D”) son los bytes que el microcontrolador necesita esperar, indican que se configuro de manera correcta. En términos de programación del microcontrolador la forma de salir de línea de comandos AT sería con la función `atcn()`;

```

atcn
OK
|
61 74 63 6E 0D
4F 4B 0D
atcn();

```

Figura 4.37: Bytes necesarios para modificar el número de red del módulo Xbee®.

4.3.7. Flujo básico de las rutinas para comunicación mediante el chip de radiofrecuencia Xbee®

El flujo básico seguido en el programa para la implementación del esquema de radiofrecuencia se muestra en la Figura 4.38.

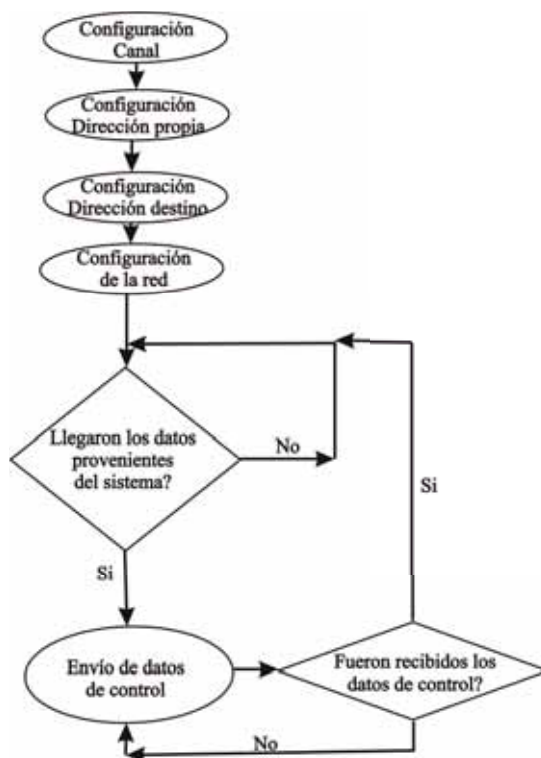


Figura 4.38: Diagrama de flujo del esquema implementado basado en el módulo Xbee®.

Como se observa en la Figura 4.38 se configura primero el canal de radiofrecuencia que se va a utilizar, la dirección del módulo Xbee® que se va a utilizar para identificarlo dentro de la red de radiofrecuencia, posteriormente se configura la dirección del módulo Xbee® con el que se quiere comunicar y la dirección de la red que se va a utilizar. Una

vez configurados estos parámetros se espera por los datos de los sensores del sistema, una vez que llegaron los datos se procesan y se envían los correspondientes datos de control los cuales son transmitidos hasta que el sistema los procese, una vez procesados se espera de nueva cuenta por los datos del sistema.

4.4. Implementación del esquema de comunicaciones

TCP/IP

Una comunicación de red (comunicación entre 2 o más computadoras) que usa TCP/IP (el cual es un protocolo orientado a conexión y con control de errores que garantiza la integridad de la información) es ideal para aplicaciones de automatización y control. Para establecer una comunicación de red utilizando TCP/IP se recurrió a las librerías de TCP/IP contenidas en el lenguaje de programación LabVIEW®.

Para realizar una comunicación TCP/IP deben existir al menos dos estaciones que van a realizar la comunicación, una estación denominada servidor la cual espera por una solicitud de conexión (provenientes de otra computadora) al número de puerto seleccionada y a la dirección IP de la computadora y una estación denominada cliente la cual es la encargada de solicitar conexión al puerto y dirección IP de la estación servidor.

4.4.1. Estación servidor TCP/IP

El bloque encargado de esperar por una solicitud de acceso al número de puerto y dirección IP para establecer la conexión TCP/IP se muestra en la Figura 4.39.

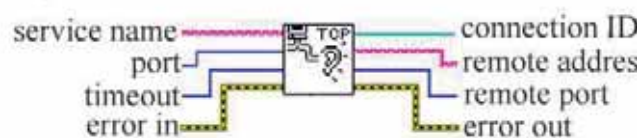


Figura 4.39: Librería TCP/IP para esperar la conexión al puerto.

Los principales conectores de esta librería son:

- **port**. Es el número del puerto que se va a utilizar para la transmisión de datos tanto la computadora servidor como la computadora cliente deben de tener en cuenta el

número de puerto que va a utilizar para realizar la transmisión de datos.

- **timeout.** Es el periodo de tiempo durante el cual se va a esperar por una conexión. Si la conexión no se cumple dentro de este periodo de tiempo se indica un error de conexión.
- **error in.** Informa sobre condiciones de error que se presentaron antes de arrancar la espera para la conexión del puerto.
- **connection ID.** Es un número de referencia de la red el cual identifica la conexión TCP. Esta es usada para posteriores llamadas de librerías de escritura y lectura de datos.
- **remote address.** Este indica la dirección IP is de la computadora cliente.
- **remote port.** Indica el número del puerto que utiliza la computadora cliente para la conexión.
- **error out.** Este nos indica de errores que se presenten durante la ejecución del programa.

4.4.2. Estación cliente TCP/IP

La estación cliente inicia una comunicación llamando a una estación servidor usando el numero IP y puerto de la estación con la cual se quiere realizar la comunicación

En la Figura 4.40 se muestra el bloque encargado abrir una conexión TCP/IP con el puerto y dirección de la computadora servidor.

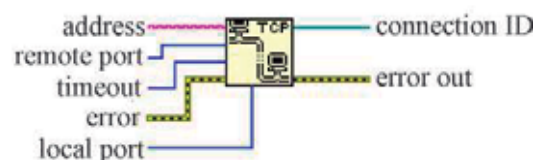


Figura 4.40: Librería TCP/IP para establecer la conexión con la computadora servidor.

Los principales conectores de esta librería son:

- **address.** Es la dirección con la que desea establecer una conexión.

- **remote port.** Es el puerto o el nombre del servicio con el que desea establecer una conexión, para realizar la transmisión de datos.
- **timeout.** es el periodo de tiempo en milisegundos de espera para que una máquina servidora conteste .
- **error.** Describe las condiciones de error que se producen antes de que este nodo se ejecute.
- **local port.** El puerto local es el puerto de conexión local. Algunos servidores sólo permiten las conexiones a los clientes que utilizan números de puerto dentro de un rango específico que depende del servidor. Si el valor es 0, el sistema operativo selecciona un puerto no utilizado.
- **connection ID.** Es el número de la conexión TCP.
- **error out.** Contiene la información de los errores presentados durante la ejecución del programa.

4.4.3. Interconexión entre la computadora cliente y servidora vía TCP/IP

Como se muestra en la Figura 4.41 se muestra como se realiza la interconexión entre 2 computadoras vía TCP/IP mediante el lenguaje de programación LabVIEW®. La estación servidora espera por una petición para acceder al puerto especificado proveniente de la estación cliente, la estación cliente usa el numero IP y puerto de la estación con la cual se quiere realizar la comunicación, en caso de que el puerto y la IP sean correctas la estación servidora responde y se establece una interconexión entre las 2 computadoras.

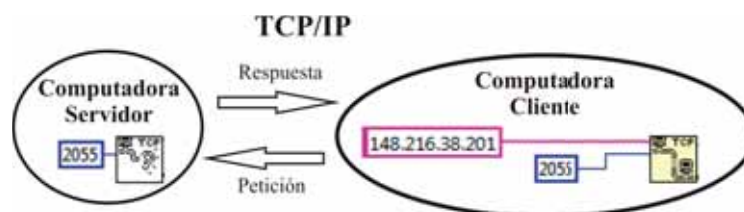


Figura 4.41: Interconexión entre la computadora cliente y servidora vía TCP/IP.

4.4.4. Librerías encargadas de transmitir, recibir datos mediante TCP/IP

Una vez establecida la conexión los datos pueden fluir de la estación cliente a la servidora y viceversa de manera continua, este flujo de información se controla con los bloques TCP Read (Figura 4.42) y TCP Write (Figura 4.43) los cuales permiten recibir y enviar datos, mediante una conexión remota.

En la Figura 4.42 se muestra el bloque encargado de transmitir datos mediante el protocolo TCP/IP.

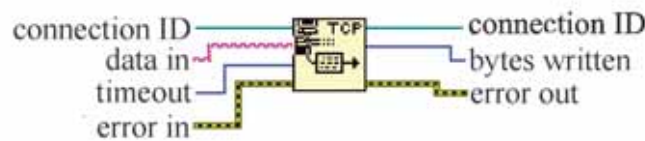


Figura 4.42: Librería TCP/IP para transmitir datos.

Los principales conectores de esta librería son:

- **connection ID.** Número de referencia de la red que identifica a la conexión TCP
- **data in.** Contiene los datos que desea escribir en la conexión.
- **timeout.** Es el período de tiempo en milisegundos de espera para la función para escribir bytes en un dispositivo antes de la función termine.
- **error in.** Describe las condiciones de error que ocurrieron antes de cerrar la conexión TCP.
- **connection ID out.** Contiene la misma información que la entrada connection ID.
- **bytes written.** Es el número de bytes que se escriben en la conexión.
- **error out.** Contiene la información de los errores presentados durante la ejecución del programa.

En la Figura 4.43 se muestra el bloque encargado de recibir datos mediante el protocolo TCP/IP el cual lee un número de bytes de una conexión TCP/IP.

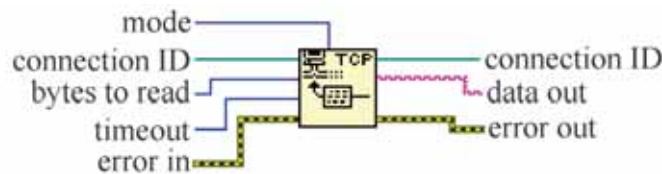


Figura 4.43: Librería TCP/IP para recibir datos.

Los principales conectores de esta librería son:

- **mode.** Indica la forma de la lectura de los datos en la conexión TCP/IP. Por default maneja una lectura estándar la cual consiste en esperar todos los bytes especificados en el conector bytes to read.
- **connection ID.** Número de referencia de la red que identifica a la conexión TCP
- **bytes to read.** es el numero de bytes a leer en el puerto.
- **timeout.** Es el período de tiempo en milisegundos de espera para la función para recibir bytes en un dispositivo antes de la función termine.
- **error in.** Describe las condiciones de error que ocurrieron antes de cerrar la conexión TCP.
- **connection ID out.** Contiene la misma información que la entrada connection ID.
- **bytes written.** Es el número de bytes que se escriben en la conexión.
- **data out.** Contiene el dato a leer provenientes de la conexión TCP/IP.
- **error out.** Contiene la información de los errores presentados durante la ejecución del programa.

4.4.4.1. Transmisión de datos vía TCP/IP

Una vez establecida la interconexión entre la computadora servidor y la computadora receptor se pueden comenzar a transmitir datos desde la computadora servidor o receptor usando las librerías TCP/IP de LabVIEW®[®], la transmisión se realiza como se muestra en la Figura 4.44.

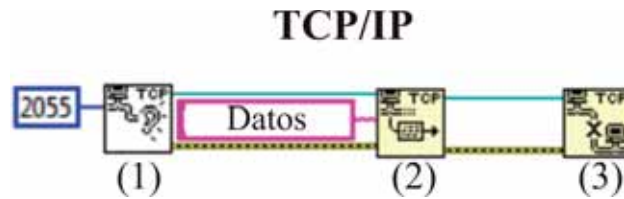


Figura 4.44: Transmisión de datos vía TCP/IP utilizando las librerías LabVIEW®.

En la Figura 4.44 se muestra la forma que se usa para la transmisión de datos que se utilizó, el bloque marcado con el número 1 es el correspondiente al que se explicó en la sección 4.4.3 el cual mantiene la interconexión entre computadoras mediante el puerto y dirección IP especificadas por la computadora servidor. El bloque marcado con el número 2 es el utilizado para realizar la transmisión de datos, los datos que se transmiten son caracteres del tipo ASCII (acrónimo inglés de American Standard Code for Information Interchange, Código Estadounidense Estándar para el Intercambio de Información). Una vez que se terminan de transmitir los datos, la interconexión debe ser cerrada el bloque marcado con el número 3 muestra la forma de cerrar la interconexión, en caso de no ser necesario transmitir o recibir más datos la sesión no debe ser cerrada.

4.4.4.2. Recepción de datos vía TCP/IP

Una vez establecida la interconexión entre la computadora servidor y la computadora receptor se pueden comenzar a transmitir datos desde la computadora servidor o receptor usando las librerías TCP/IP de LabVIEW®, la recepción se realiza como se muestra en la Figura 4.45.

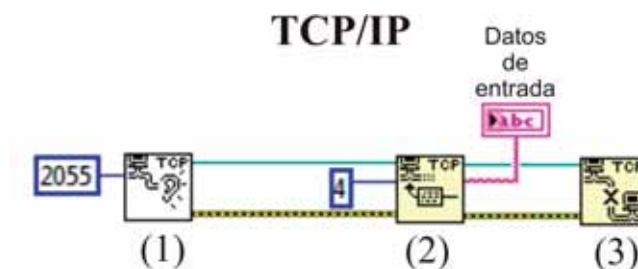


Figura 4.45: Recepción de datos vía TCP/IP utilizando las librerías LabVIEW®.

En la Figura 4.45 se muestra la forma de recibir datos, el bloque marcado con el número 1 es el correspondiente al explicado en la sección 4.4.3 el cual mantiene la interconexión entre computadoras mediante el puerto y dirección IP especificadas por la computadora servidor. El bloque marcado con el número 2 es utilizado para la recepción

de datos, primero es necesario determinar el número de datos que se van a recibir (los datos recibidos son cadenas de bytes), posteriormente una vez que se reciben los datos que se reciben son caracteres del tipo ASCII. Una vez que se termina de recibir los datos, la interconexión debe ser cerrada el bloque marcado con el número 3 muestra la forma de cerrar la interconexión, en caso de no ser necesario transmitir o recibir más datos la sesión no debe ser cerrada.

4.4.5. Librería encargada cerrar la conexión TCP/IP

En la Figura 4.46 se muestra el bloque encargado de cerrar la conexión TCP/IP.

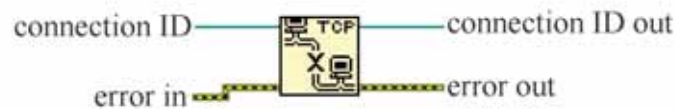


Figura 4.46: Librería TCP/IP para cerrar la conexión.

Los principales conectores de esta librería son:

- **connection ID.** Es el número de la conexión TCP que se quiere cerrar.
- **error in.** Describe las condiciones de error que ocurrieron antes de cerrar la conexión TCP.
- **connection ID out.** Contiene la misma información que la entrada connection ID.
- **error out.** Contiene la información de los errores presentados durante la ejecución del programa.

4.5. Implementación del esquema de comunicaciones TELNET

LabVIEW® cuenta con librerías TELNET para enviar y recibir datos a través de una conexión TELNET. El protocolo TELNET es un protocolo de Internet estándar que permite conectar terminales y aplicaciones en Internet. Al igual que en el protocolo TCP/IP, TELNET proporciona reglas básicas que permiten vincular a un cliente con un intérprete de comandos (servidor). Para establecer una comunicación de red utilizando

TELNET se recurre a las librerías de TELNET contenidas en el lenguaje de programación LabVIEW®.

Para realizar una comunicación TELNET deben existir al menos dos estaciones que van a realizar la comunicación, una estación denominada servidor la cual espera por una solicitud de conexión (provenientes de otra computadora) al número de puerto seleccionada y a la dirección IP de la computadora y una estación denominada cliente la cual es la encargada de solicitar conexión al puerto y dirección IP de la estación servidor.

4.5.1. Estación servidor TELNET

El bloque encargado de esperar por una solicitud de acceso al número de puerto y dirección IP para establecer la conexión TELNET se muestra en la Figura 4.47.

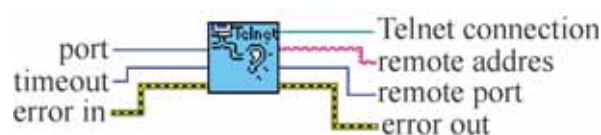


Figura 4.47: Librería TELNET para esperar la conexión al puerto.

Los principales conectores de esta librería son:

- **port**. Es el número del puerto que se va a utilizar para la transmisión de datos tanto la computadora servidor como la computadora cliente deben de tener en cuenta el número de puerto que va a utilizar para realizar la transmisión de datos.
- **timeout**. Es el periodo de tiempo durante el cual se va a esperar por una conexión. Si la conexión no se cumple dentro de este periodo de tiempo se indica un error de conexión.
- **error in**. Informa sobre condiciones de error que se presentaron antes de arrancar la espera para la conexión del puerto.
- **TELNET connection**. Es un número de referencia para la conexión TELNET. Esta es usada para posteriores llamadas de librerías de escritura y lectura de datos.
- **remote address**. Este indica la dirección IP is de la computadora cliente.
- **remote port**. Indica el número del puerto que utiliza la computadora cliente para la conexión.

- **error out.** Este nos indica de errores que se presenten durante la ejecución del programa.

4.5.2. Estación cliente TELNET

La estación cliente inicia una comunicación llamando a una estación servidor usando el numero IP y puerto de la estación con la cual se quiere realizar la comunicación.

En la Figura 4.48 se muestra el bloque encargado abrir una conexión TELNET con el puerto y dirección de la computadora servidor.

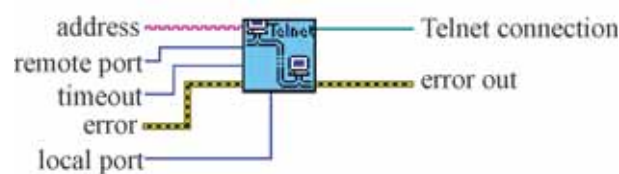


Figura 4.48: Librería TELNET para solicitar acceso a la computadora servidor.

Los principales conectores de esta librería son:

- **address.** Es la dirección con la que desea establecer una conexión.
- **remote port.** Es el puerto o el nombre del servicio con el que desea establecer una conexión, para realizar la transmisión de datos.
- **timeout.** es el periodo de tiempo en milisegundos de espera para que una máquina servidora conteste .
- **error.** Describe las condiciones de error que se producen antes de que este nodo se ejecute.
- **local port.** El puerto local es el puerto de conexión local. Algunos servidores sólo permiten las conexiones a los clientes que utilizan números de puerto dentro de un rango específico que depende del servidor. Si el valor es 0, el sistema operativo selecciona un puerto no utilizado.
- **connection ID.** Es el número de la conexión TCP.
- **error out.** Contiene la información de los errores presentados durante la ejecución del programa.

4.5.3. Interconexión entre la computadora cliente y servidora vía TELNET

Como se muestra en la Figura 4.49 se muestra como se realiza la interconexión entre 2 computadoras vía TELNET mediante el lenguaje de programación LabVIEW®.

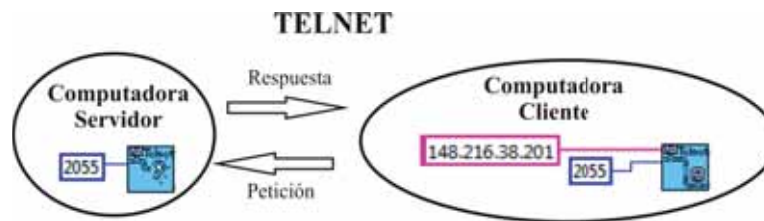


Figura 4.49: Interconexión entre la computadora cliente y servidora vía TELNET.

La estación servidora espera por una petición para acceder al puerto especificado proveniente de la estación cliente, la estación cliente usa el número IP y puerto de la estación con la cual se quiere realizar la comunicación, en caso de que el puerto y la IP sean correctas la estación servidora responde y se establece una interconexión entre las 2 computadoras.

4.5.4. Librerías encargadas de transmitir, recibir datos mediante TELNET

Una vez establecida la conexión los datos pueden fluir de la estación cliente a la servidora y viceversa de manera continua, este flujo de información se controla con los bloques TELNET Read (Figura 4.50) y TELNET Write (Figura 4.51) los cuales permiten recibir y enviar datos, mediante una conexión remota.

En la Figura 4.50 se muestra el bloque encargado de transmitir datos mediante el protocolo TELNET.

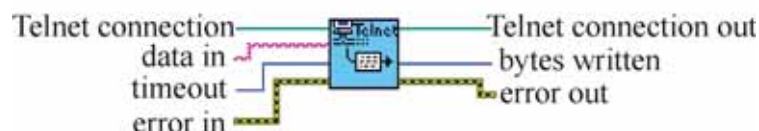


Figura 4.50: Librería TELNET para transmitir datos.

Los principales conectores de esta librería son:

- **TELNET connection.** Es un número de referencia para la conexión TELNET.
- **data in.** Contiene los datos que desea escribir en la conexión.
- **timeout.** Es el período de tiempo en milisegundos de espera para la función para escribir bytes en un dispositivo antes de la función termine.
- **error in.** Describe las condiciones de error que ocurrieron antes de cerrar la conexión TELNET.
- **TELNET connection out.** Contiene la misma información que la entrada TELNET connection.
- **bytes written.** Es el número de bytes que se escriben en la conexión.
- **error out.** Contiene la información de los errores presentados durante la ejecución del programa.

En la Figura 4.51 se muestra el bloque encargado de recibir datos mediante el protocolo TELNET el cual lee un número de bytes de una conexión TELNET.

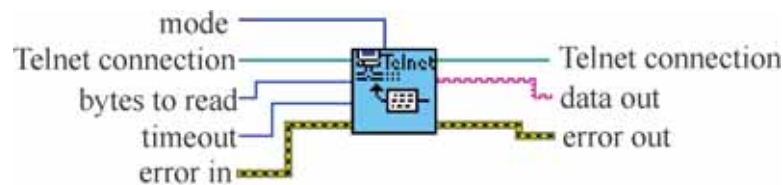


Figura 4.51: Librería TELNET para recibir datos.

Los principales conectores de esta librería son:

- **mode.** Indica la forma de la lectura de los datos en la conexión TELNET. Por default maneja una lectura estándar la cual consiste en esperar todos los bytes especificados en el conector bytes to read.
- **TELNET connection.** Es un número de referencia para la conexión TELNET.
- **bytes to read.** es el numero de bytes a leer en el puerto.
- **timeout.** Es el período de tiempo en milisegundos de espera para la función para recibir bytes en un dispositivo antes de la función termine.

- **error in.** Describe las condiciones de error que ocurrieron antes de cerrar la conexión TELNET.
- **TELNET connection out.** Contiene la misma información que la entrada TELNET connection.
- **bytes written.** Es el número de bytes que se escriben en la conexión.
- **data out.** Contiene el dato a leer provenientes de la conexión TELNET.
- **error out.** Contiene la información de los errores presentados durante la ejecución del programa.

4.5.4.1. Transmisión de datos vía TELNET

Una vez establecida la interconexión entre la computadora servidor y la computadora receptor se pueden comenzar a transmitir datos desde la computadora servidor o receptor usando las librerías TELNET de LabVIEW®. la transmisión se realiza como se muestra en la Figura 4.52.

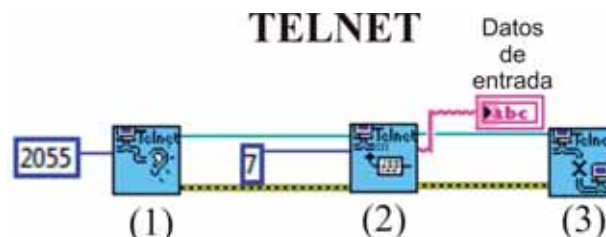


Figura 4.52: Transmisión de datos vía TELNET utilizando las librerías LabVIEW®.

En la Figura 4.52 se muestra la forma que se usa para la transmisión de datos que se utilizó, el bloque marcado con el número 1 es el correspondiente al que se explicó en la sección 4.4.3 el cual mantiene la interconexión entre computadoras mediante el puerto y dirección IP especificadas por la computadora servidor. El bloque marcado con el número 2 es el utilizado para realizar la transmisión de datos, los datos que se transmiten son caracteres del tipo ASCII. Una vez que se terminan de transmitir los datos, la interconexión debe ser cerrada el bloque marcado con el número 3 muestra la forma de cerrar la interconexión, en caso de no ser necesario transmitir o recibir más datos la sesión no debe ser cerrada.

4.5.4.2. Recepción de datos vía TCP/IP

Una vez establecida la interconexión entre la computadora servidor y la computadora receptor se pueden comenzar a transmitir datos desde la computadora servidor o receptor usando las librerías TELNET de LabVIEW®, la recepción se realiza como se muestra en la Figura 4.53.

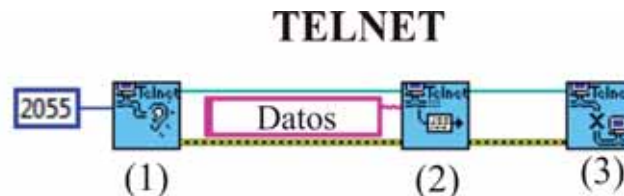


Figura 4.53: Recepción de datos vía TELNET utilizando las librerías LabVIEW®.

En la Figura 4.53 se muestra la forma de recibir datos, el bloque marcado con el número 1 es el correspondiente al explicado en la sección 4.4.3 el cual mantiene la interconexión entre computadoras mediante el puerto y dirección IP especificadas por la computadora servidor. El bloque marcado con el número 2 es utilizado para la recepción de datos, primero es necesario determinar el número de datos que se van a recibir (los datos recibidos son cadenas de bytes), posteriormente una vez que se reciben los datos que se reciben son caracteres del tipo ASCII. Una vez que se termina de recibir los datos, la interconexión debe ser cerrada el bloque marcado con el número 3 muestra la forma de cerrar la interconexión, en caso de no ser necesario transmitir o recibir más datos la sesión no debe ser cerrada.

4.5.5. Librería encargada cerrar la conexión TELNET

En la Figura 4.54 se muestra el bloque encargado de cerrar la conexión TELNET.

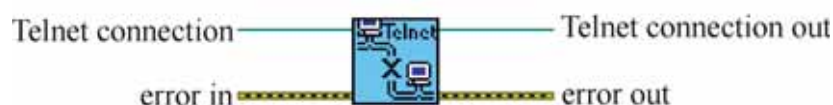


Figura 4.54: Librería TELNET para cerrar la conexión.

Los principales conectores de esta librería son:

- **TELNET connection.** Es el número de la conexión TELNET que se quiere cerrar.

- **error in.** Describe las condiciones de error que ocurrieron antes de cerrar la conexión TELNET.
- **TELNET connection out.** Contiene la misma información que la entrada TELNET connection.
- **error out.** Contiene la información de los errores presentados durante la ejecución del programa.

4.6. Flujo básico de las rutinas para comunicación TCP/IP y TELNET en LabVIEW®

Se puede decir que hay dos flujos básicos para la realización de una rutina que soporte la comunicación TCP/IP y TELNET en LabVIEW®, el establecimiento de la comunicación y el intercambio de datos, a continuación se presenta un diagrama y una breve explicación de cada uno.

4.6.1. Establecimiento de la comunicación

Para poder establecer una comunicación entre la computadora servidor y la computadora cliente, la computadora servidor espera el tiempo establecido a que la computadora cliente inicie una comunicación por el puerto especificado, durante este tiempo la computadora cliente debe llamar a la computadora servidor y solicitar la conexión por un puerto especificado, si la computadora servidor detecta que una conexión se está solicitando a través del puerto establecido se establece la comunicación y tanto el cliente como el servidor quedan en capacidad para enviar y recibir datos.

En la Figura 4.55 se aprecia el diagrama de flujo de conexión en la computadora servidor.

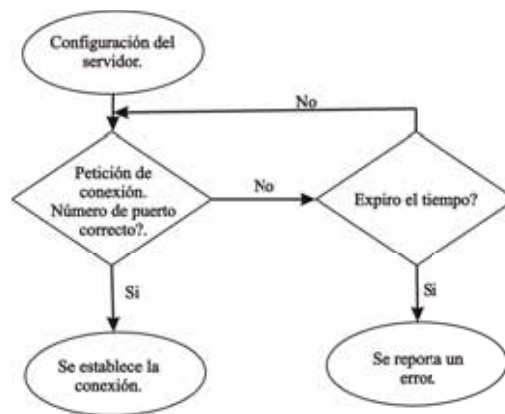


Figura 4.55: Diagrama de flujo del establecimiento de conexión en la computadora servidor.

En la Figura 4.56 se pueden apreciar los diagramas de flujo de conexión en la computadora cliente.

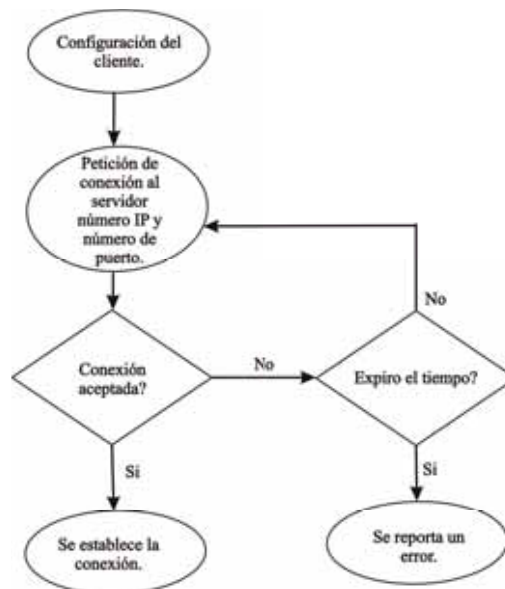


Figura 4.56: Diagrama de flujo del establecimiento de conexión en la computadora cliente.

Los mismos diagramas de flujo descritos fueron usados tanto para la interconexión vía TCP/IP y la interconexión TELNET.

4.6.2. Envío de datos o escritura

Cualquiera de las dos estaciones envía un número de datos determinado, si la estación receptora no recibe todos los datos se reporta un error en la estación emisora, en la Figura 4.57 se muestra el diagrama de flujo correspondiente a la transmisión de datos vía

TCP/IP o TELNET y en la Figura 4.58 se muestra el diagrama de flujo correspondiente a la recepción de datos.

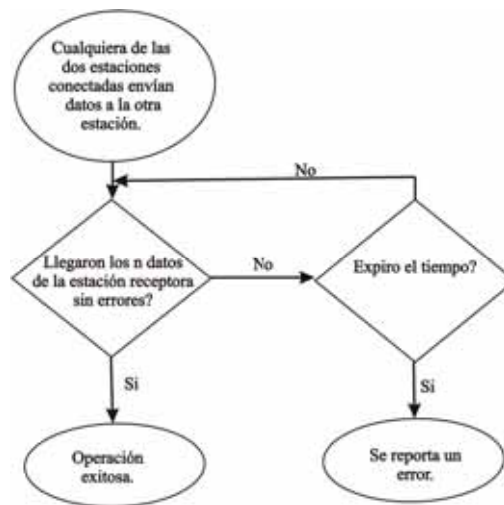


Figura 4.57: Diagrama de flujo de la transmisión de datos vía TCP/IP o TELNET.

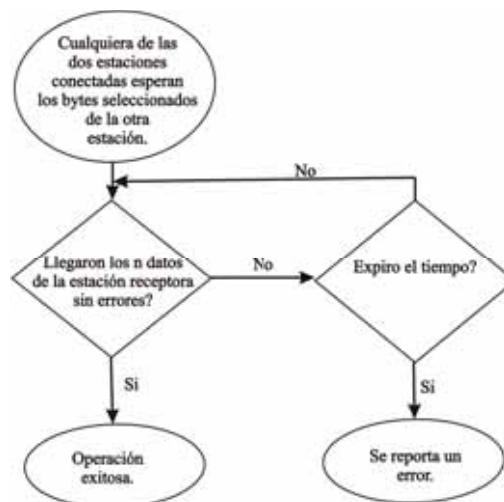


Figura 4.58: Diagrama de flujo de la recepción de datos vía TCP/IP ó TELNET.

4.6.3. Diagrama de conexión del sistema mediante el esquema TCP/IP

Una vez que se tienen las funciones para transmitir y recibir datos el diagrama de conexión del sistema seguido para el monitoreo y control del sistema se muestra en la Figura 4.59.



Figura 4.59: Monitoreo y control del sistema mediante la conexión TCP/IP.

El sistema de nivel de líquido se conecta como se muestra en la Figura. 4.59. El monitoreo y control se realiza desde una computadora remota dentro de la red de computadoras del laboratorio la cual sirve como la computadora cliente, el sistema de nivel de líquidos tiene una computadora conectada la cual sirve como la computadora servidor con un puerto habilitado para la transferencia de datos. La computadora servidora espera la petición de la computadora cliente una vez que la conexión se realiza con éxito la computadora cliente tiene el control del sistema ya que la computadora servidora tiene conectado mediante el puerto RS232 la tarjeta basada en el PIC18F2550 (descrita en la sección 4.2) la cual tiene acceso a las electroválvulas, sensores y bomba de agua del sistema.

4.6.4. Diagrama de conexión del sistema mediante el esquema TELNET

Una vez que se tienen las funciones para transmitir y recibir datos el diagrama de conexión del sistema seguido para el monitoreo y control del sistema se muestra en la Figura 4.60.



Figura 4.60: Monitoreo y control del sistema mediante la conexión TCP/IP.

El sistema de nivel de líquido se conecta como se muestra en la Figura. 4.60. El monitoreo y control se realiza desde una computadora remota la cual sirve como la computadora cliente, el sistema de nivel de líquidos tiene una computadora conectada la cual sirve

como la computadora servidor con un puerto habilitado para la transferencia de datos. La computadora servidora espera la petición de la computadora cliente una vez que la conexión se realizó con éxito la computadora cliente tiene el control del sistema ya que la computadora servidor tiene conectado mediante el puerto RS232 la tarjeta basada en el pic18F2550 (descrita en la sección 4.2) la cual tiene acceso a las electroválvulas, sensores y bomba de agua del sistema.

4.7. Implementación del esquema de comunicaciones GSM

El sistema global para las comunicaciones móviles GSM es un sistema estándar, de telefonía móvil digital. Un cliente GSM puede conectarse a través de su teléfono con su computadora o cualquier otro sistema que se requiera y enviar y recibir mensajes por correo electrónico, faxes, navegar por Internet, acceder con seguridad a Internet, así como utilizar otras funciones digitales de transmisión de datos, incluyendo el servicio de mensajes cortos (SMS).

Durante el capítulo 2 se habló sobre el funcionamiento del esquema GSM. En esta sección se explica la manera en la que se aplicó el esquema de comunicaciones GSM en el sistema de nivel de líquido, mediante el uso del módulo GSM/GPRS SM5100B fabricado por LinkSprite Technologies Inc. El sistema se controla de manera manual mediante la transmisión de mensajes cortos (SMS) los cuales en caso de tener la estructura adecuada pueden modificar cualquiera de las variables del sistema como modificar el voltaje que se le aplica a las electroválvulas y el encendido de la bomba de agua, así como la transmisión de un mensaje el cual indica el nivel de líquido en cada tanque del sistema.

4.7.1. Módulo SM5100B

El módulo seleccionado para la conexión del sistema de nivel de líquido mediante GSM fue el módulo GSM SM5100B fabricado por LinkSprite Technologies Inc. (Figura 4.61) cuatribanda (GSM-850, EGSM- 900, DCS-1800, PCS-1900), el cual por sus características nos permite integrarlo en un gran número de proyectos inalámbricos. Este módulo tiene acceso a casi cualquier cosa que un teléfono móvil normal puede como lo son los mensajes de texto SMS, transmisión de datos vía GPRS, transmisión de datos vía TCP/IP, entre

otras aplicaciones [Manual SM5100B, 2006].

Las principales características con las que cuenta este módulo son:

- Rango de temperatura: -10 a 55 ° C (en funcionamiento), -40 a 85 ° C (almacenamiento).
- Conexión: el acceso de sus pines es mediante un conector de 60 pines.
- Fuente de alimentación: 3.3 volts a 4.2 volts, voltaje de CD. 3.6 voltaje típico.
- Consumo de energía:
 - ▷ Modo de apagado: <100uA.
 - ▷ El modo de reposo: <2.0mA.
 - ▷ Modo en espera: <7.0mA.
 - ▷ Modo de comunicación: 350 mA (para la banda GSM).
 - ▷ Modo de comunicación: 2 A (para la banda GSM).
- Bandas de Frecuencia:
 - ▷ GSM850.
 - ▷ EGSM900.
 - ▷ DCS1800.
 - ▷ PCS1900.
- Potencia de transmisión:
 - ▷ Clase 4 (2W) para EGSM900/GSM850.
 - ▷ Clase 1 (1W) para DCS1800/PCS1900.
- Puerto UART0 con flujo de control con velocidad de hasta 460Kbps.
- Puerto UART1 sin control de flujo.
- Dos convertidores analógicos a digitales de 10 bits.
- Entradas y salidas disponibles para un altavoz y un micrófono para la recepción o transmisión de llamadas de voz en caso de ser necesario.

- Memoria flash de 32 Megabit (2Mb x 16-Bit).
- Memoria RAM de 4 megabits (256 x 16-Bit).
- Reconocimiento automático de la SIM.
- Dimensiones:
 - ▷ 35.0 x 39.0 x 2.9mm.
 - ▷ Peso: <9g.



Figura 4.61: Aspecto real del módulo GSM SM5100B.

En la Figura 4.61 se puede observar el aspecto real del módulo GSM SM5100B en su lado superior e inferior. Para poder tener acceso a los pines del módulo GSM es necesario recurrir a un conector de 60 pines (Figura 4.62), la separación entre cada pin es de 0.5mm. En la Figura 4.62 se puede observar el conector de 60 pines en su lado superior e inferior [Sparkfun, 2011].

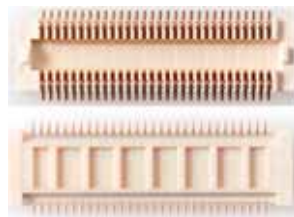


Figura 4.62: Conector de 60 pines para acceder a los pines del módulo SM5100B.

El uso de la tarjeta SIM (acrónimo en inglés de subscriber identity module, en español módulo de identificación del suscriptor) es obligatorio en las redes GSM. Para que el módulo pueda acceder a cualquier servicio GSM es necesario tener acceso a un chip SIM, la cual es una tarjeta inteligente desmontable usada en teléfonos móviles almacenan de forma segura la clave de servicio del suscriptor usada para identificarse ante la red GSM.

En la Figura 4.63 se muestra el conector que utiliza el módulo para tener acceso a la tarjeta SIM [Sparkfun, 2011].



Figura 4.63: Aspecto real del conector SIM.

El conector SIM (Figura 4.63) es un conector de 6-pines de conexión estándar de montaje superficial.

El módulo no necesita directamente de una antena para tener acceso a la señal GSM sin embargo para mejorar cuestiones de cobertura se recurrió a la selección de la antena que se muestra en la Figura 4.64 [Sparkfun, 2011].



Figura 4.64: Aspecto real de la antena cuatri-banda para el módulo GSM.

La antena mostrada en la Figura 4.64 es una antena celular que nos permite manejar las bandas:

- GSM/850E: 824 a 894MHz.
- GSM: 880 a 960MHz.
- DCS: 1710 a 1880MHz.
- PCS: de 1850 a 1990MHz.

También tiene un conector SMA (SubMiniature version A) estándar, este tipo de conector es un tipo de conector roscado para cable coaxial utilizado para señales de microondas, es útil en frecuencias de hasta 33GHz. Este tipo de antena seleccionada nos ayuda a mejorar la cobertura del módulo GSM.

Para implementar directamente el conector SIM y el conector de 60 pines en el módulo GSM se recurrió a la tarjeta mostrada en la Figura 4.65 [Sparkfun, 2011].

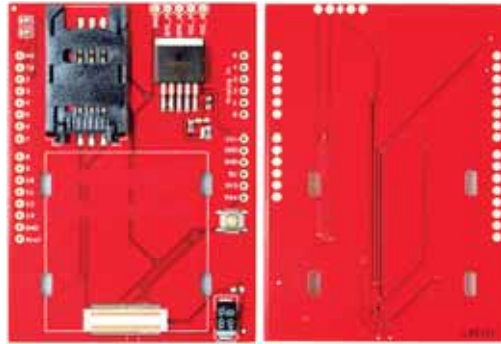


Figura 4.65: Tarjeta utilizada para adquirir las señales del módulo GSM.

La tarjeta mostrada en la Figura 4.65 nos permite montar el módulo GSM, montar el chip SIM para que el módulo GSM adquiera los datos del SIM, contiene 2 pines de salida para adquirir las señales provenientes de los puertos UART0 y UART1 y un regulador de voltaje para regular el voltaje a 3.8 volts. Esta tarjeta nos permitió tener acceso de las señales de salida del módulo y tener el control del módulo directamente con el microcontrolador PIC18F2550.

Dadas las características del módulo GSM SM5100B nos permite controlarlos mediante dispositivos que manejen directamente un puerto de comunicación serial asíncrona (como se muestra en la Figura. 4.66 y la Figura. 4.67) . A través de este puerto, el módulo GSM SM5100B puede comunicarse con cualquier dispositivo o puerto con comunicación UART (por ejemplo algún microcontrolador o algún puerto RS232 de una computadora).



Figura 4.66: Esquema general de conexión entre una computadora y un microcontrolador mediante un módulo GSM SM5100B.



Figura 4.67: Esquema general de conexión entre un microcontrolador y el módulo GSM SM5100B.

La comunicación serial dependerá de que los 2 puertos UART (el del microcontrolador y el del módulo GSM) sean conjurados con opciones compatibles (la velocidad de transmisión, la paridad, el bit de inicio, el bit de paro, y los bits de datos), por default el módulo GSM viene configurado con una velocidad de 9600 bps y su forma de programación es mediante comandos AT. Para esta tesis el módulo GSM que se utilizó se configuró para transmitir y recibir mensajes SMS, así como transmitir y recibir llamadas desde cualquier celular.

4.7.1.1. Modo de comandos AT

Para modificar o leer los parámetros del módulo GSM, el módulo debe ser configurado mediante comandos AT. Cada comando AT comienza con los caracteres AT (caracteres hexadecimales 0x41, 0x54) y termina con el <CR> (carácter hexadecimal 0x0D). El conjunto de comandos AT implementados por el módulo SM5100B se pueden dividir en dos categorías: "básico" y "extendida".

En la estructura básica los comandos AT tienen el formato de "AT <x> <CR>", donde "<x>" es el comando que se va a introducir, y <CR> es el carácter de final que indica el final del comando AT [Comandos AT del módulo SM5100B, 2008].

En la estructura extendida para los comandos AT, estos comandos pueden funcionar en varios modos, como se muestra en la Tabla 4.6.

Tabla 4.6: Formato de los comandos AT en su forma extendida para el módulo GSM 5100B-D.

Tipo de Comando.	Sintaxis.	Ejemplo
Comando de lectura.	AT+<x>=?	AT+CMEE=?
Comando de escritura.	AT+<x>?	AT+CMEE?
Comando de escritora (con parametro de entrada).	AT+<x>=<...>	AT+CMEE=0
Comando de escritora (sin parametro de entrada).	AT+<x>	AT+CGSN

Al introducir un comando en el módulo se tendrán diferentes respuestas:

- o La respuesta de cada comando ejecutado comienza y termina con el carácter <LF> (carácter hexadecimal 0x0A).

- Si la sintaxis del comando ingresada es incorrecta, una cadena de "error" serán devueltos.
- Si en la sintaxis del comando es correcto, pero de transmisión con parámetros erróneos, la cadena de error "+ CME:<error>" o "+ ERROR CMS:<error>" serán devueltos.
- Si un comando AT se ha ejecutado con éxito, una cadena "OK" se devuelve.

En la Figura 4.68 se muestra un ejemplo que muestra la introducción de comandos AT en su forma extendida. La introducción del comando se observa de color azul una vez introducidos los comandos de manera correcta el módulo responde con una cadena de confirmación la cual está representada por los bytes de color rojo.

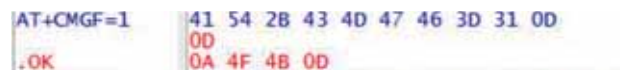


Figura 4.68: Forma de introducción de comandos AT en el módulo GSM.

4.7.2. Tarjeta inalámbrica para el control del módulo GSM/GPRS SM5100B basada en el microcontrolador PIC18F2550

En la Figura 4.69 se muestra el diagrama de bloques de las tarjetas inalámbricas diseñada en base al microcontrolador PIC18F2550 y el módulo GSM/GPRS SM5100B.

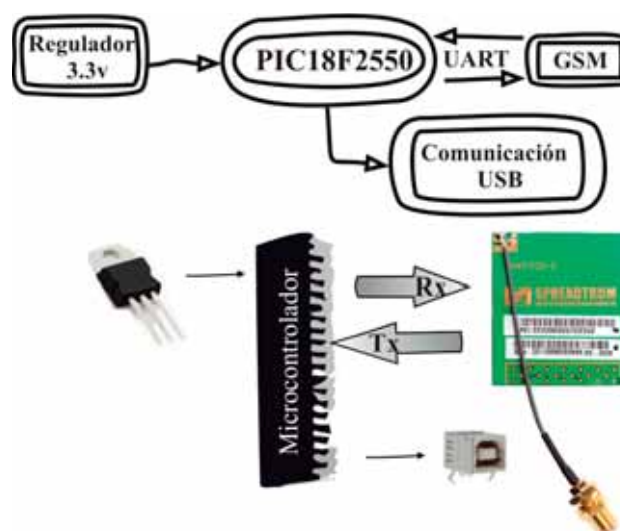


Figura 4.69: Diagrama de bloques de las tarjetas inalámbricas basadas en el módulo GSM/GPRS SM5100B.

La tarjeta que se diseñó en base al microcontrolador PIC18F2550 (Figura 4.70) está pensada para habilitar todos los pines del microcontrolador, con sus correspondientes conectores de alimentación para alimentar el microcontrolador y el módulo GSM, la habilitación del puerto USART para comunicarnos con el módulo GPRS, la habilitación del puerto USB para proyectos futuros y un LED indicador que nos indica el funcionamiento de la tableta.

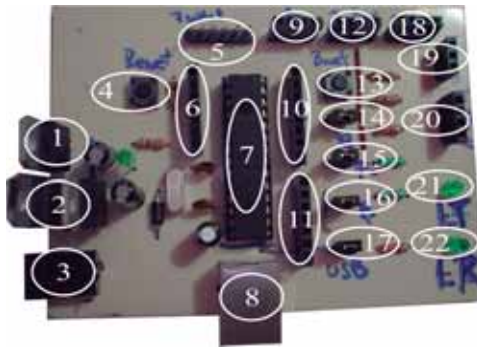


Figura 4.70: Aspecto real de la tarjeta diseñada basada en el PIC18F2550 para el control del módulo GSM.

En la Figura 4.70 se muestra el aspecto real de la tarjeta diseñada para el tener al control del módulo GSM. En la Figura 4.70 se enumeran las diferentes características del circuito diseñado, las cuales son descritas a continuación.

1. Se tiene un regulador LD33V el cual por su característica el voltaje mínimo de entrada debe ser de 8 volts y permite regular una entrada de hasta 35 volts de cd para obtener a la salida 3.3 volts con los que se puede usar para alimentar algún dispositivo en especial a este voltaje, la salida máxima de corriente es de .5A a 5v.
2. Se tienen dos reguladores LM7805 el cual por su característica el voltaje mínimo de entrada debe ser de 8 volts y permite regular una entrada de hasta 35 volts de cd para obtener a la salida 5 volts con los que se alimenta el microcontrolador y el módulo GSM, la salida máxima de corriente es de 1A a 5v por regulador necesitamos en condiciones extremas hasta 2 ampéres por lo que se colocaron 2 reguladores para soportar la demanda de corriente que se pueda llegar a tener.
3. La entrada de voltaje de corriente directa se hace mediante un conector tipo jack el cual permite colocar cualquier eliminador de CD para alimentar la tarjeta.
4. Este push button permite reiniciar el microcontrolador.

5. Este adaptador permite colocar algún programador externo (ICD ó PICKIT) en caso de ser necesario programar la memoria de programa del microcontrolador.
6. Este bus de salida permite acceder al puerto A del microcontrolador para adquirir o transmitir señales, la corriente de salida de cada pin del microcontrolador es de hasta 20mA.
7. Este conector permite quitar o colocar otro microcontrolador en caso de que se dañe.
8. Este conector permite programar el microcontrolador vía USB o bien transmitir datos desde el microcontrolador a la computadora.
9. Estos buses de salida nos permiten acceder a la tierra (GND).
10. Este bus de salida permite acceder al puerto B del microcontrolador para adquirir o transmitir señales, la corriente de salida de cada pin del microcontrolador es de hasta 20mA.
11. Este bus de salida permite acceder al puerto B del microcontrolador para adquirir o transmitir señales, la corriente de salida de cada pin del microcontrolador es de hasta 20mA.
12. Este bus de salida permite acceder a los 3.3volts del regulador LD33V.
13. Este push button permite junto con el push button para reiniciar el microcontrolador (componente 4) programar el microcontrolador vía USB.
14. Este conector tipo jumper permite deshabilitar la programación del microcontrolador vía USB en caso de realizar la programación por algún programador externo como el PICKIT o el ICD (programadores de MICROCHIP®).
15. Este conector tipo jumper permite deshabilitar la recepción de paquetes de información del microcontrolador al módulo GSM.
16. Este conector tipo jumper permite deshabilitar la transmisión de paquetes de información del microcontrolador al módulo GSM.
17. Este conector tipo jumper permite deshabilitar la alimentación del de la tarjeta vía USB, la alimentación de voltaje máximo por el puerto USB es de .5A a 5volts.

18. Este bus de salida permiten acceder a los 5volts del regulador LM7805.
19. Este bus de salida permiten alimentar con 5volts el módulo GSM.
20. Este bus de salida permite tener acceso directo al pin de transmisión serial y recepción serial para conectar directamente al módulo GSM.
21. Cuando el módulo GSM está en operación este indicador tipo LED indica en cuando algún tipo de paquetes con información son recibidos.
22. Cuando el módulo GSM está en operación este indicador tipo LED indica cuando algún tipo de paquetes con información son transmitidos desde el microcontrolador para configurar el módulo GSM.

La forma de conexión que se siguió se muestra en la Figura 4.71 se muestra el esquema general seguido para el control manual de las electroválvulas, adquisición de las señales de los sensores de ultrasonido, la señal de control de la bomba de agua y la transmisión de datos vía UART para transmitir todos los datos vía GSM.

En la Figura 4.71 se puede observar que el microcontrolador es el encargado de adquirir y transmitir los datos del sistema mediante el módulo GSM. Mediante un mensaje de texto el operador del sistema puede solicitar la altura de los tanques del sistema esto mediante la lectura de las señales provenientes de la instrumentación de los sensores de ultrasonido que miden el nivel de líquido, también mediante un mensaje de texto el controlador puede modificar el voltaje que se entrega a las electroválvulas, apagar o encender la bomba de agua y solicitar la altura de los tanques. Esto nos permite tener el control y monitoreo de todas las variables del sistema de manera manual desde cualquier celular.



Figura 4.71: Esquema general de conexión del sistema mediante la tarjeta basada en el módulo GSM.

4.7.3. Funciones programadas

Se programaron funciones para configurar el módulo GSM, transmitir llamadas, recibir llamadas, transmitir mensajes SMS y recibir mensajes SMS, las cuales nos permiten tener el control así como configurar el módulo GSM.

4.7.3.1. Funciones para configurar el módulo GSM

Al conectar el módulo GSM, Tomará alrededor de quince a treinta segundos para que el módulo se inicialice en su totalidad. En la Figura 4.72 se muestra la trama de inicialización del módulo GSM.

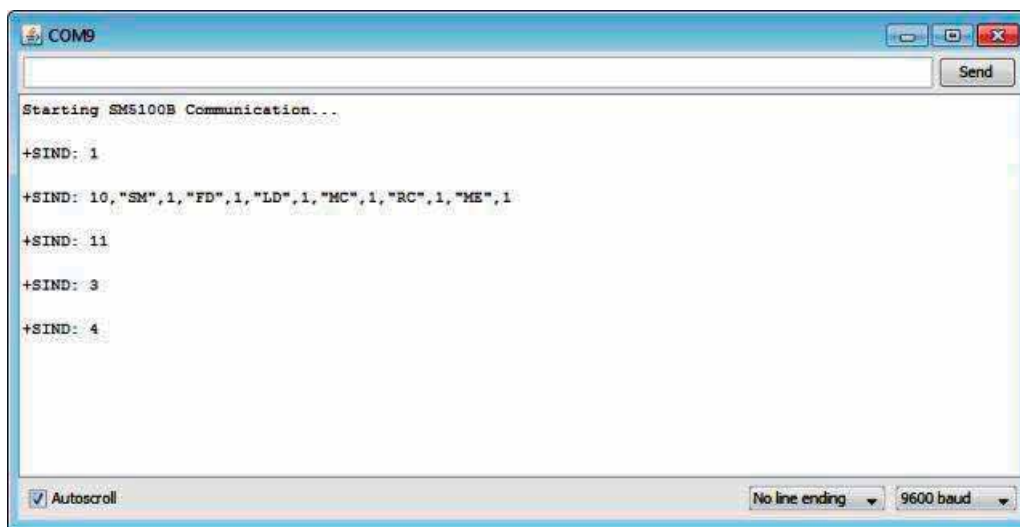


Figura 4.72: Trama de inicialización del módulo GSM.

Lo que se presenta en la Figura 4.72 es un registro de las acciones del módulo GSM al ser inicializado. La trama de inicialización se puede interpretar de la siguiente manera [Comandos AT del módulo SM5100B, 2008]:

- + SIND: 1 significa que la tarjeta SIM se ha insertado, en caso de que el módulo no encuentre la tarjeta SIM este registro mandara un valor de 0.
- + SIND: 10 este parámetro le indica al módulo GSM que muestre el estado de la red telefónica en el módulo, viene seguida de la trama que se observa en la Figura 4.72 la cual indica el estado del módulo durante la conexión, en ocasiones viene acompañada de la cadena "+STIN:0" el cual indica el estado de la SIM.
- + SIND: 11 este parámetro significa que el módulo GSM se ha registrado en la red celular.

- + SIND: 3 este parámetro significa que el módulo está parcialmente listo para comunicarse.
- + SIND: 4 este parámetro indica que el módulo está registrado en la red, y listo para comunicarse, en caso de presentarse un error este parámetro será indicado con un 8 el cual nos indica que la conexión del módulo con la red GSM no se pudo realizar.

Para poder usar el módulo de manera adecuada se tiene que esperar el registro de caracteres ASCII “+ SIND: 4” el cual nos permite determinar que el módulo se inicializo de manera adecuada, en caso de presentarse un error el registro de caracteres ASCII que se presentara será “+ SIND: 8”, en caso de tener este registro será un indicador de tener que inicializar el módulo GSM. A partir de que el módulo se encuentra inicializado de manera adecuada se pueden ingresar los comandos AT necesarios para configurar el módulo GSM, transmitir llamadas, recibir llamadas, transmitir mensajes SMS y recibir mensajes SMS.

La Tabla 4.7 muestra los comandos AT necesarios para configurar la hora, la fecha del módulo GSM [Comandos AT del módulo SM5100B, 2008].

Tabla 4.7: Lista de comandos AT encargados de la configuración de la fecha y hora del módulo GSM.

Comando.	Nombre y descripción.
<i>AT+CCLK</i>	Permite leer o escribir la hora y fecha del módulo. Comando de escritura: $AT+CCLK="yy/mm/dd,hh:mm:ss"$ Comando de lectura: $AT+CCLK?$

Como se observa en la la Tabla 4.7 para el comando de escritura de fecha y hora del módulo se tiene que escribir la cadena ASCII “yy/mm/dd,hh:mm:ss” en donde “yy” representa el año, “mm” representa el mes, “dd” representa el día, “hh” representa la hora, “mm” representan los minutos y “ss” representan los segundos. En la Figura 4.73 se muestra la manera de introducir la fecha y hora en el módulo.

```

.....
.+SIND: 1 0A 2B 53 49 4E 44 3A 20 31 0D
.
.+SIND: 0A 2B 53 49 4E 44 3A 20 31 30 2C 22
10,"SM",1,"F 53 4D 22 2C 31 2C 22 46 44 22 2C 31
D",1,"LD",1, 2C 22 4C 44 22 2C 31 2C 22 4D 43 22
"MC",1,"RC", 2C 31 2C 22 52 43 22 2C 31 2C 22 4D
1,"ME",1 45 22 2C 31 0D
.
.+STIN:0 0A 2B 53 54 49 4E 3A 30 0D
.
.+SIND: 11 0A 2B 53 49 4E 44 3A 20 31 31 0D
.
.+SIND: 3 0A 2B 53 49 4E 44 3A 20 33 0D
.
.+SIND: 4 0A 2B 53 49 4E 44 3A 20 34 0D
.AT+CCLK="11 0A 41 54 2B 43 43 4C 4B 3D 22 31 31
/06/22,18:47 2F 30 36 2F 32 32 2C 31 38 3A 34 37
:00" 3A 30 30 22 0D
.
.OK 0D
. 0A 4F 4B 0D
. 0A

```

Figura 4.73: Configuración de la fecha y hora en el módulo GSM.

En la Figura 4.73 se observa que una vez que el módulo se inicializo de manera adecuada se transmiten los parámetros de fecha y hora (bytes de azul). Una vez que los datos fueron introducidos de manera adecuada el módulo responderá con una cadena ASCII “0D,0A,4F,4B,0D,0A”(últimos bytes marcados de rojo).

La Tabla 4.8 muestra los comandos AT necesarios para configurar la frecuencia del módulo GSM [Comandos AT del módulo SM5100B, 2008].

Tabla 4.8: Lista de comandos AT encargados de la configuración de la frecuencia del módulo GSM.

Comando.	Nombre y descripción.	Parámetro.
AT+SBAND	Permite seleccionar la frecuencia del módulo. Comando de escritura: <i>AT+CCLK=parámetro</i> Comando de lectura: <i>AT+SBAND?</i>	0=GSM900 1=DCS1800 2=PCS1900 3=GSM850 4=GSM900-DCS1800 5=GSM850-GSM900 6=GSM850-DCS1800 7=GSM850-PCS1900 8=GSM900-PCS1900 9=GSM850-GSM900-DCS1800 10=GSM850-GSM900-PCS1900

De acuerdo a la empresa que da el servicio GSM y la región se tienen las frecuencias de operación en este caso el proveedor trabaja con frecuencias de 850MHz y 1900MHz por lo que en este caso se selecciono la banda 7 para configurar el módulo la cual nos permitirá tener acceso a los servicios GSM. En la Figura 4.74 se muestra la manera de modificar la banda en el módulo GSM.

```

.....
.+SIND: 1      D8 D8 D8 D8 D8 0D
               0A 2B 53 49 4E 44 3A 20 31 0D
               0A 0D
.+SIND:        0A 2B 53 49 4E 44 3A 20 31 30 2C 22
10,"SM",1,"F   53 4D 22 2C 31 2C 22 46 44 22 2C 31
D",1,"LD",1,    2C 22 4C 44 22 2C 31 2C 22 4D 43 22
"MC",1,"RC",    2C 31 2C 22 52 43 22 2C 31 2C 22 4D
1,"ME",1        45 22 2C 31 0D
               0A 0D
.+STIN:0       0A 2B 53 54 49 4E 3A 30 0D
               0A 0D
.+SIND: 11     0A 2B 53 49 4E 44 3A 20 31 31 0D
               0A 0D
.+SIND: 3      0A 2B 53 49 4E 44 3A 20 33 0D
               0A 0D
.+SIND: 4      0A 2B 53 49 4E 44 3A 20 34 0D
.AT+SBAND=7    0A 41 54 2B 53 42 41 4E 44 3D 37 0D
               0D
               0D
.OK           0A 4F 4B 0D
.|           0A

```

Figura 4.74: Configuración de la frecuencia de trabajo del módulo GSM.

En la Figura 4.74 se observa que una vez que el módulo se inicializo de manera adecuada se transmiten los parámetros de frecuencia (bytes de azul). Una vez que los datos fueron introducidos de manera adecuada el módulo responderá con una cadena ASCII “0D,0A,4F,4B,0D,0A”(últimos bytes marcados de rojo). La función programada para modificar la frecuencia es “sband(x)” donde x es el valor de frecuencia que se a utilizar.

4.7.3.2. Función para transmitir llamadas

A partir de que el módulo se encuentra inicializado de manera adecuada y trabajando en la frecuencia adecuada se pueden ingresar los comandos AT necesarios para transmitir llamadas a un determinado número telefónico. La Tabla 4.9 muestra los comandos AT necesarios para realizar o colgar una llamada telefónica [Comandos AT del módulo SM5100B, 2008].

Tabla 4.9: Lista de comandos AT encargados de la transmisión y finalización de una llamada telefónica en del módulo GSM.

Comando.	Nombre y descripción.	Ejemplo.
ATD	Permite generar una llamada a un número telefónico específico. Comando de escritura: <i>ATD=Número telefónico</i>	<i>ATD4431998817</i>
ATH	Permite finalizar una llamada telefónica. Comando de escritura: <i>ATH</i>	<i>ATH</i>
ATA	Permite contestar una llamada telefónica. Comando de escritura: <i>ATA</i>	<i>ATA</i>

El modulo tiene entradas para micrófono y bocinas los cuales nos permitirían escuchar y transmitir voz a través del módulo sin embargo para el caso de este trabajo se dejo hasta la habilitación del servicio de llamadas para transmitir o recepción de señales que indiquen algún evento por lo que no se utiliza la habilitación de las bocinas o micrófonos transmisión o recepción de voz. En la Figura 4.75 se muestra la manera de introducir el número telefónico una vez que ya se ha inicializado el módulo.

```

.....
.+SIND: 1 0A D9 D9 D9 D9 D9 0D
          0A 2B 53 49 4E 44 3A 20 31 0D
          0A 0D
          0A 2B 53 49 4E 44 3A 20 31 30 2C 22
10,"SM",1,"F 53 4D 22 2C 31 2C 22 46 44 22 2C 31
D",1,"LD",1, 2C 22 4C 44 22 2C 31 2C 22 4D 43 22
"MC",1,"RC", 2C 31 2C 22 52 43 22 2C 31 2C 22 4D
1,"ME",1, 45 22 2C 31 0D
          0A 0D
          0A 2B 53 54 49 4E 3A 30 0D
.+STIN:0 0A 0D
          0A 2B 53 49 4E 44 3A 20 31 31 0D
.+SIND: 11 0A 0D
          0A 2B 53 49 4E 44 3A 20 33 0D
.+SIND: 3 0A 0D
          0A 2B 53 49 4E 44 3A 20 34 0D
.+SIND: 4 0A 2B 53 49 4E 44 3A 20 34 0D
.ATD44319988 0A 41 54 44 34 34 33 31 39 39 38 38
17 31 37 0D
          0D
          0A 2B 53 49 4E 44 3A 20 35 2C 31 0D
.+SIND: 5,1 0A 0D
          0A 2B 53 49 4E 44 3A 20 32 0D
.+SIND: 2 0A 2B 53 49 4E 44 3A 20 32 0D
.ATH 0A 41 54 48 0D
          0D
          0A 4F 4B 0D
          0A

```

Figura 4.75: Transmisión de llamadas con el módulo GSM.

En la Figura 4.75 se observa que una vez que el módulo se inicializo de manera adecuada se establece la llamada mediante el comando ATD y el número del teléfono con el que nos queremos comunicar, posteriormente si el módulo no establece conexión con el número que se determinó se regresara una cadena ASCII “NO CARRIER” o en caso de presentarse un error se presentara una cadena ASCII “ERROR”, cuando la realización de la llamada es correcta el módulo responde con las cadenas “+SIND: 5,1” y “+SIND: 2” las cuales indican que la llamada se está llevando a cabo. Una vez que se quiere terminar la llamada se introduce el comando ATH y al terminar la conexión el módulo responderá con una cadena ASCII “0D,0A,4F,4B,0D,0A”. Las funciones programadas para la transmisión de llamadas desde el microcontrolador fueron ini_llamada() la cual nos permite transmitir una llamada a un número específico y la función ter_llamada() la cual nos permite mandar llamar el parámetro ATH y terminar la llamada [Comandos AT del módulo SM5100B, 2008].

4.7.3.3. Función para transmitir mensajes SMS

Para poder transmitir o recibir mensajes de texto es necesario configurar el módulo de manera adecuada para habilitar estas funciones. La Tabla 4.10 muestra los comandos AT necesarios para configurar el módulo para transmitir o recibir mensajes vía SMS [Comandos AT del módulo SM5100B, 2008].

Tabla 4.10: Lista de comandos AT encargados de configurar, transmitir y recibir mensajes de texto SMS.

Comando.	Nombre y descripción.	Ejemplo.
<i>AT+CMGF</i>	Permite determinar la forma en la que se va a general un mensaje a un número telefónico específico. Comando de escritura: <i>AT+CMGF=parámetro</i> Comando de lectura: <i>AT+CMGF=?</i>	0=Modo PDU 1=Modo texto
<i>AT+CNMI</i>	Permite controlar la manera de indicar cuando se recibe un mensaje de texto. Comando de escritura: <i>AT+CNMI= valor1, valor2, valor3, valor4</i> Comando de lectura: <i>AT+CNMI=?</i>	valor 1: permite determinar el modo de recepción de mensajes, el valor por default es 3. valor 2: coloca la forma de indicar los mensajes en el módulo, el valor por default es 3. valor 3: configura la forma de guardar los mensajes, el valor por default es 0. valor 4: configura la forma de indicar la situación del mensaje, el valor por default es de 0. (Se utilizaron los valores por default)
<i>AT+CMGS</i>	Permite transmitir un mensaje de texto. Comando de escritura: <i>AT+CMGS="Número" >mensaje (ctrl+z/ESC)</i>	-

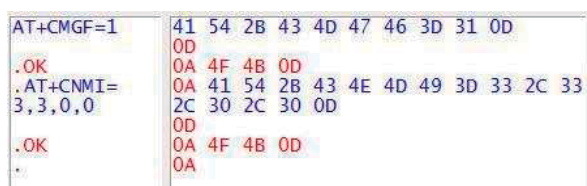
Hay 2 tipos de modos de transmisión de datos vía SMS, el primero es el modo PDU y el segundo es el modo de texto.

4.7.3.3.1. Modo de texto SMS Este es un sistema desarrollado en 1992 que permite la transmisión bidireccional de mensajes de texto desde celulares, dispositivos GSM conectados a un computador y páginas de Internet. Usa el formato de texto ASCII. La longitud de transmisión de caracteres es máxima de 160 caracteres ASCII [Dreamfebric 2006].

4.7.3.3.2. Modo PDU SMS El modo de transmisión PDU utiliza tramas en formato de paquetes para enviar mensajes SMS a través de la red GSM. Los datos se transmiten mediante una cadena de caracteres codificados en hexadecimal. La principal diferencia de este modo de transmisión al de mensajería corta es que debido a su formato estándar, la información que transmite entrega información acerca del usuario, centro servidor, datos de transmisión entre otros.

Para esta tesis se selecciono el método de transmisión de datos mediante mensajería corta SMS debido a que se mandan mensajes con caracteres ASCII para hacer el control manual y no es necesaria la información que se puede llegar a obtener con un mensaje vía PDU [Dreamfebric 2006].

4.7.3.3.3. Configuración del módulo GSM para la transmisión de mensajes de texto En la Figura 4.76 se muestra la manera de configurar el módulo para la transmisión de mensajes mediante el modo texto. Se introduce primero el comando `AT+CMGF=1` con el cual le indicamos al módulo GSM que se va hacer una transmisión de datos en modo texto, el módulo responderá con una cadena ASCII `"0D,0A,4F,4B,0D,0A"` la cual nos indica que el módulo fue configurado de manera adecuada. Posteriormente se introduce el comando `AT+CNMI=3,3,0,0` con sus valores por default el cual nos permite controlar la manera de indicar cuándo se recibe un mensaje de texto y de igual manera el módulo responderá con una cadena ASCII `"0D,0A,4F,4B,0D,0A"` esta configuración le dice al módulo GSM que envíe de inmediato cualquier dato SMS al puerto serie.



```

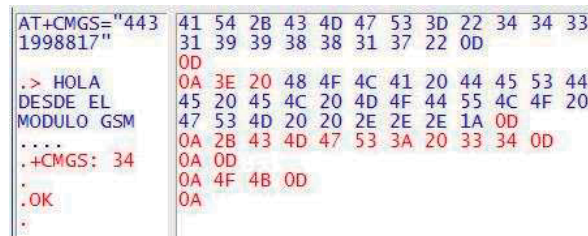
AT+CMGF=1      41 54 2B 43 4D 47 46 3D 31 0D
                0D
                0A 4F 4B 0D
                0A 41 54 2B 43 4E 4D 49 3D 33 2C 33
AT+CNMI=3,3,0,0 2C 30 2C 30 0D
                0D
                0A 4F 4B 0D
                0A

```

Figura 4.76: Configuración del módulo GSM para la transmisión de mensajes de texto.

Una vez que se encuentra el módulo configurado se recurre al comando `AT+CMGS`, en la Figura 4.77 se observa la forma de transmisión de datos. Una vez que se introduce el comando y el número telefónico con el que se va a comunicar el módulo responde con la cadena ASCII `"0D,0A,3E,20"` la cual es la que indica que se puede introducir la información a transmitir una vez que se tiene la información que se va a transmitir se tiene que introducir el comando (`ctrl+z/ESC`) el cual corresponde al número hexadecimal `1A` el

cual indica que los datos deben ser transmitidos cuando el módulo termina la transmisión responderá con una cadena ASCII “0D, 0A, 2B, 43, 4D, 47, 53, 3A, 20, 33, 34, 0D, 0A, 0D, 0A, 4F, 4B, 0D, 0A”.



```

AT+CMGS="443
1998817"
41 54 2B 43 4D 47 53 3D 22 34 34 33
31 39 39 38 38 31 37 22 0D
0D
.> HOLA
0A 3E 20 48 4F 4C 41 20 44 45 53 44
DESDE EL
45 20 45 4C 20 4D 4F 44 55 4C 4F 20
MODULO GSM
47 53 4D 20 20 2E 2E 2E 1A 0D
....
0A 2B 43 4D 47 53 3A 20 33 34 0D
.CMGS: 34
0A 0D
0A 4F 4B 0D
.OK
0A
.
```

Figura 4.77: Transmisión de mensajes de texto.

Las funciones `cmgfi()` y `cnmi()` nos permiten configurar el módulo para transmitir mensajes, la función `cmgs()` nos permite transmitir el mensaje que se desea a un número específico [Comandos AT del módulo SM5100B, 2008].

4.7.3.4. Recepción de llamadas

Una vez que se configuro el módulo con la frecuencia adecuada (como se explico en la sección 4.7.3.3.3) el módulo GSM está listo para recibir llamadas. Cuando el módulo GSM recibe una llamada de otro número, el módulo devuelve el texto como se muestra en la Figura 4.78. Por cada llamada se muestra que aparece la cadena de caracteres ASCII “RING” seguida de la cadena “+CLIP: “Número del usuario que está llamando al módulo”,129”. Estas cadenas son las que son detectadas para realizar la recepción de llamadas. El módulo repetirá la cadena hasta que el módulo conteste la llamada con el comando “ATA” o hasta que el usuario que está llamando cancele la llamada.



```

.RING
.+CLIP: "4431024486",129
.RING
.+CLIP: "4431024486",129
.RING
.+CLIP: "4431024486",129
.RING
.+CLIP: "4431024486",129
.RING
.+CLIP: "4431024486",129
.RING
.+CLIP: "4431024486",129
.
```

Figura 4.78: Recepción de llamadas en el módulo GSM.

4.7.3.5. Recepción de mensajes SMS

Una vez que se configuro el módulo (como se explico en la sección 4.7.3.2) el módulo está listo para recibir mensajes SMS. Cuando el módulo GSM recibe un mensaje de otro número, el módulo devuelve el texto como se muestra en la Figura 4.79. Por cada mensaje se muestra que aparece la cadena de caracteres ASCII de “+CMT” seguida del número remitente, el número 529410000141 es el número del SMSC (centro de servicio de mensajes cortos). Luego tenemos la fecha y hora. La línea que comienza con el carácter ASCII “#” y termina con el carácter ASCII “9” es el contenido del mensaje. Estas cadenas son las que son detectadas para realizar la recepción de mensajes SMS.

Figura 4.79

Figura 4.79: Recepción de mensajes SMS en el módulo GSM.

4.7.4. Flujo básico de las rutinas para comunicación mediante comunicación GSM

El flujo básico seguido en el programa para la implementación del esquema mediante la comunicación GSM se muestra en la Figura 4.80. La idea que se siguió es que el módulo y el microcontrolador estén conectados directamente en el sistema y mediante un mensaje de texto de un celular definido se modifican los parámetros del sistema.



Figura 4.80: Diagrama de flujo del esquema implementado basado en esquema GSM.

Como se observa en la Figura 4.80 se configura el módulo GSM para transmitir, recibir

mensajes y se configura la banda de frecuencia. Una vez configurados estos parámetros el microcontrolador espera por los datos (cadena “#b0e00e00”) en un mensaje de texto para modificar el voltaje de las electroválvulas y bomba del sistema, una vez que el sistema recibe los parámetros adecuados, el microcontrolador envía las alturas de los tanques en un mensaje SMS, para que el usuario pueda monitorear el sistema ante su acción de control en lazo abierto.

4.7.5. Recepción de mensajes de texto

En la figura 4.81 se muestra los bytes que se tienen al momento de recibir un mensaje de texto el inicio de la cadena viene dada por los bytes “+CMT:” seguido de los bytes del número de la fuente del mensaje que en este caso es “4431998817”, después de los bytes que contienen el número se tienen 55 bytes con toda la información referente al mensaje de entrada, entre los datos más importantes que tenemos son la hora (byte 29, byte 30), minutos (byte 32, byte 33), segundos (byte 35, byte 36), año (byte 20, byte 21), mes (byte 23, byte 24), día (byte 26, byte 27), y la información del mensaje (byte 45 [#, byte identificador], byte 46 [b, bomba], byte 47 [estado bomba], byte 48 [e, electroválvula 1], byte 49 [voltaje electroválvula 1], byte 50 [voltaje electroválvula 1], byte 51 [e, electroválvula 2], byte 52 [voltaje electroválvula 2], byte 53[voltaje electroválvula 2]). En la Tabla 6.1 se muestra la información



Figura 4.81: Recepción de mensajes de texto en el módulo GSM.

En la Tabla 4.11 se muestra la estructura de los 9 bytes del mensaje seleccionado para controlar el sistema de nivel de líquido.

Tabla 4.11: Estructura del mensaje seleccionado para controlar el sistema de nivel de líquido.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
#	b	0	e	0	0	e	0	0

El byte 1 es un simple byte que nos indica el inicio del mensaje para realizar el control, en caso de que este byte sea diferente el microcontrolador descarta el mensaje completo y espera por un mensaje nuevo. El byte 2 indica que se va a introducir un valor en la bomba de agua y el byte 3 indica el estado de la bomba de agua que se quiere en la Tabla 4.12 se muestran los posibles valores y estados que se pueden presentar.

Tabla 4.12: Estructura del mensaje para controlar la bomba de agua del sistema de nivel de líquido.

Byte 2	Valor byte 3	Estado de la bomba
b	0	Apagado
	1	Prendido
	otro	Apagado

El byte 4 indica se va a introducir un valor de voltaje en la electroválvula 1 y el byte 5 y byte 6 nos indican el voltaje que se va a introducir en la electroválvula 1 en la Tabla 4.13 se muestran los posibles valores que se pueden presentar.

Tabla 4.13: Estructura del mensaje para controlar la electroválvula 1 del sistema de nivel de líquido.

Byte 4	Valor byte 5	Valor byte 6	Voltaje de la electroválvula 1
e	0	0	0
	0	1	1
	0	2	2
	0	3	3
	0	4	4
	0	5	5
	0	6	6
	0	7	7
	0	8	8
	0	9	9
	1	0	10
	otro	otro	10

En caso de tener parámetros en el mensaje fuera del rango o que no cumplen con los rangos el microcontrolador abrirá al máximo la electroválvula 1 para evitar que se pueda llegar a presentar un derrame de nivel de líquido.

El byte 7 indica se va a introducir un valor de voltaje en la electroválvula 2 y el byte 8 y byte 9 nos indican el voltaje que se va a introducir en la electroválvula 2 en la Tabla 4.14 se muestran los posibles valores que se pueden presentar.

Tabla 4.14: Estructura del mensaje para controlar la electroválvula 2 del sistema de nivel de líquido.

Byte 7	Valor byte 8	Valor byte 9	Voltaje de la electroválvula 2
e	0	0	0
	0	1	1
	0	2	2
	0	3	3
	0	4	4
	0	5	5
	0	6	6
	0	7	7
	0	8	8
	0	9	9
	1	0	10
	otro	otro	10

En caso de tener parámetros en el mensaje fuera del rango o que no cumplen con los rangos el microcontrolador abrirá al máximo la electroválvula 2 para evitar que se pueda llegar a presentar un derrame de nivel de líquido.

4.8. Implementación del controlador PID

A lo largo del capítulo 4 se describen los métodos que se aplicaron para el monitoreo remoto del sistema de nivel de líquido, para la parte de control del sistema se implemento un controlador PID utilizando la librería de control de LabVIEW® (Figura 4.82).

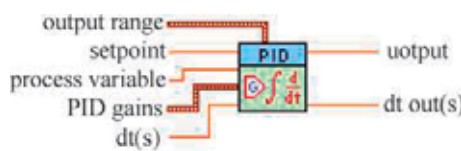


Figura 4.82: Librería PID implementada para el control del sistema.

Los principales conectores de esta librería son:

- **output range.** Especifica el rango al que se va a forzar la salida de controlador.

Este conector consta de 2 parámetros:

- ▷ **output high.** Especifica el valor máximo de la salida del controlador.
- ▷ **output low.** Especifica el valor mínimo de la salida del controlador.
- **setpoint.** Especifica el valor de la señal, o el valor deseado, de la variable que está siendo controlado.
- **process variable.** Especifica el valor medido de la variable de proceso que está siendo controlada. Este valor es igual al valor de realimentación del circuito.
- **PID gains.** Especifica las ganancias K_c , T_i y T_d del controlador.
 - ▷ **proportional gain (K_c).** Especifica la ganancia proporcional del controlador K_c representa la ganancia proporcional.
 - ▷ **integral time (T_i , min).** Especifica el tiempo integral en minutos.
 - ▷ **derivative time (T_d , min):** especifica el tiempo derivativo en minutos.
- **dt(s).** Especifica el intervalo, en segundos, en la que esta librería es llamada. Si dt (s) es menor o igual a cero, esta librería calcula el tiempo desde su última llamada mediante un temporizador interno con una resolución de un milisegundo.
- **output.** Devuelve la salida del algoritmo PID que se aplica al proceso controlado.
- **dt out(s).** Devuelve el intervalo de tiempo real en segundos.

Para utilizar esta librería se introdujo en el conector `outputrange` como valor máximo el valor de apertura máximo de las electroválvulas en voltaje el cual es de 10 volts y como valor mínimo el valor mínimo de apertura de las electroválvulas en voltaje el cual es de 2 volts. En el conector `setpoint` el valor en voltaje de la altura deseada del tanque a controlar. El conector `process variable` se selecciona el valor de retroalimentación del sistema es decir el voltaje actual en el tanque que se está controlando. En el conector `PID gains` se introducen las ganancias del controlador PID [Barajas, 2010].

$$K_p = 1.15.$$

$$T_i = 20\text{seg.}$$

$$T_d = 5\text{seg.}$$

En la entrada `dt(s)`, no se coloca valor, por lo que se trabajara con el valor default de LabVIEW® el cual es aproximadamente de 500 milisegundo. La salida que se toma

para mandar la señal al microcontrolador es la proveniente de la salida output la cual es procesada a un valor de 0 a 1023 para transmitirla al microcontrolador y actué sobre la electroválvula.

Capítulo 5

Diseño de las Interfaces Gráficas

Este capítulo muestra las interfaces gráficas realizadas en LabVIEW®. Se selecciona esta plataforma ya que nos permite implementar y obtener los datos de las diferentes pruebas del controlador PID implementado en lazo cerrado sobre el tanque 1 así como realizar el monitoreo y control de los esquemas de radiofrecuencia, TCP/IP y TELNET.

5.1. Bytes a transmitir y recibir necesarios para comunicarse con el sistema

En la Tabla 5.1 se muestra la estructura de la cadena de bytes recibida por el microcontrolador

Tabla 5.1: Estructura de la cadena de bytes transmitida por el microcontrolador.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
0x38	ADC0 HIGH	ADC0 LOW	ADC1 HIGH	ADC1 LOW

Los 5 bytes recibidos contienen la información del valor de voltaje de los sensores de ultrasonido. El byte 1 es el byte (numero hexadecimal 0x38) de inicio de la trama de datos el cual nos permite identificar que una trama de entrada se está recibiendo, posteriormente a este byte se tienen 4 bytes correspondientes a la parte alta y parte baja del canal 0 (voltaje leído del sensor de ultrasonido 1) y canal 1 (voltaje leído del sensor de ultrasonido 2) del convertidor analógico a digital del microcontrolador. El byte 2 corresponde a la parte alta del canal 0 del microcontrolador, el byte 3 corresponde a la parte baja del canal 0, el byte 4 corresponde a la parte alta del canal 1 y el byte 5

corresponde a la parte baja del canal 1.

A su vez el microcontrolador debe de recibir una trama de 5 bytes los cuales le indiquen que valor de voltaje se necesita aplicar a las electroválvulas y el estado de la bomba en la Tabla 5.2 se muestra el conjunto de bytes que se tienen que transmitir al microcontrolador.

Tabla 5.2: Estructura de la cadena de bytes recibida por el microcontrolador.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
PWM1 HIGH	PWM1 LOW	PWM2 HIGH	PWM2 LOW	ESTADO DE LA BOMBA

El byte 1 corresponde a la parte alta de la señal de salida PWM 1 del microcontrolador (las señales PWM utilizadas son señales de 10 bits), el byte 2 corresponde a la parte baja de la señal de salida PWM 1, el byte 3 corresponde a la parte alta de la señal de salida PWM 2, el byte 4 corresponde a la parte baja de la señal de salida PWM 2 y el byte 5 corresponde al estado de la bomba (valor hexadecimal para encender la bomba=0x01 y valor hexadecimal para apagar la bomba=0x00).

Independientemente de esquema a utilizar (TCP/IP, TELNET y Radio frecuencia) estas son las estructuras de bytes que se tienen que transmitir o leer para interactuar con el sistema.

5.2. Interfaz mediante el protocolo ZigBee

La interfaz gráfica diseñada está basada en la idea de poder controlar y monitorear de manera manual, así como también poder supervisar mediante un controlador PID el sistema, el controlador se aplica a un sistema de un solo tanque. La instrumentación se realiza mediante las tarjetas inalámbricas basadas en el microcontrolador PIC18F2550 o el microcontrolador PIC18F4550 y el módulo Xbee® que se mostraron en el capítulo anterior en la Figura 4.22. y la Figura 4.25. En la Figura 5.1 y Figura 5.2 se observa el esquema general seguido para el monitoreo y control del sistema mediante las tarjetas inalámbricas. Tanto como en el esquema de la Figura 5.1 como en el esquema de la Figura 5.2 se observa que la interfaz gráfica permite conectarse mediante el puerto RS232 con la tarjeta inalámbrica RS232 para transmitir y recibir a la tarjeta inalámbrica basada en el PIC18F4550 y la tarjeta PIC18F2550 los parámetros de control del sistema. En la Figura 5.3, Figura 5.5 y Figura 5.8 se observa el aspecto de la interfaz gráfica diseñada para este fin.

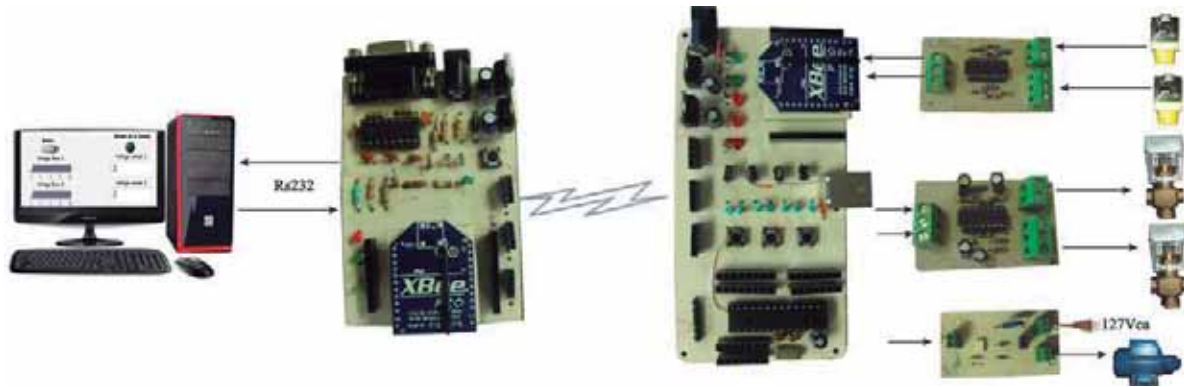


Figura 5.1: Esquema para el monitoreo y control con el módulo Xbee y el microcontrolador PIC18F2550.

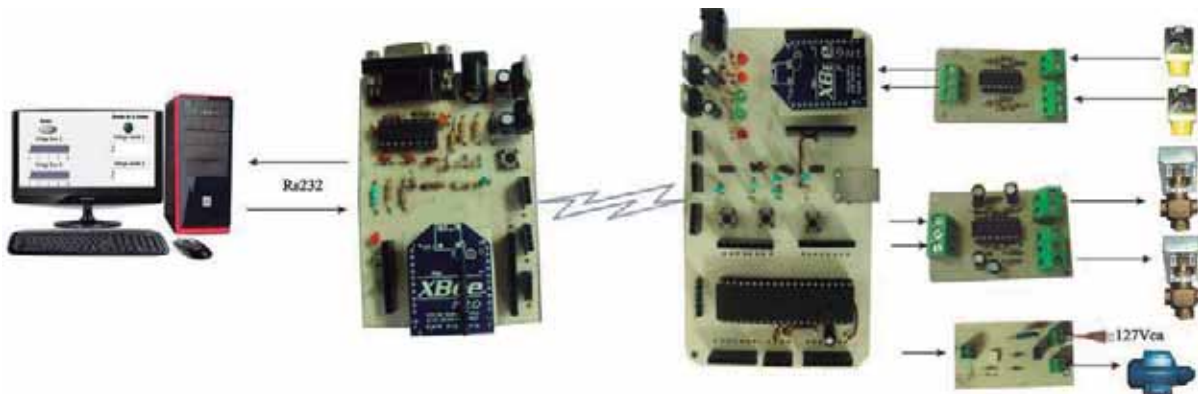


Figura 5.2: Esquema para el monitoreo y control con el módulo Xbee y el microcontrolador PIC18F4550.

La interfaz gráfica diseñada consta de 3 apartados, un apartado de configuración que nos permite seleccionar el puerto COM RS232 del módulo Xbee® conectado a la computadora, así como modificar las características de comunicación como la velocidad de transferencia de datos bytes de entrada entre otros del módulo a utilizar, un apartado manual que permite el monitoreo y control manual del sistema, y un apartado de control el cual permite observar la evolución del controlador PID en el sistema.

5.2.1. Modo configuración de la interfaz mediante el esquema de radiofrecuencia

A continuación se describe la interfaz gráfica en su modo de configuración a partir de la numeración propuesta en la Figura 5.3.

1. El microcontrolador transmite a la computadora un arreglo de 5 bytes con la información de los sensores de ultrasonido. Este apartado permite ver la cadena de datos

entrante.

2. Estos dos controladores permiten seleccionar el puerto que el microcontrolador está usando para la recepción y transferencia de datos.
3. Esta sección permite configurar todo lo referente a las propiedades de comunicación con el módulo como la velocidad de transferencia de datos el bit de paridad, el bit de paro, el control de flujo y el número de bits de datos. No es necesario configurar esta sección ya que por default se encuentra configurada.
4. Esta sección permite observar la cadena de datos recibidos, la cadena leída en forma hexadecimal, el byte número 1 de la cadena leída debe corresponder al hexadecimal 38, la cadena que se va a transmitir y el número de bytes que se van a leer provenientes del microcontrolador en este caso son 5. No es necesario configurar esta sección ya que ya se encuentra configurada.

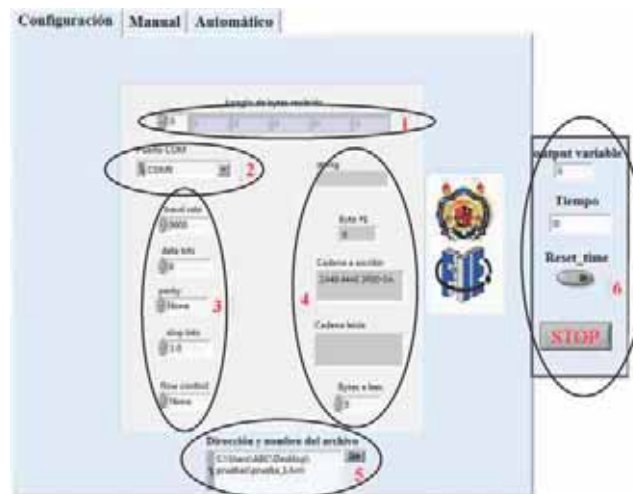


Figura 5.3: Aspecto de la interfaz gráfica diseñada para el esquema de radio frecuencia en su modo configuración.

5. Esta sección permite seleccionar la dirección y el nombre del archivo en donde se guardaran todos los datos de altura y tiempo de cada uno de los tanques del sistema. Este archivo es un archivo de texto el cual puede ser leído con programas como excel, matlab, o algún procesador de texto.
6. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el pro-

grama en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset_tiempo.

5.2.1.1. Diagrama de bloques del modo configuración de la interfaz mediante el esquema de radiofrecuencia

En la Figura 5.4 se muestra el diagrama de bloques correspondiente al modo de configuración. La parte seleccionada con el número 1 es la encargada de configurar el puerto serial COM que nos permite comunicarnos con la tarjeta Xbee RS232, esta sección es la encargada de configurar las velocidades y la transmisión y recepción de datos, el conjunto de bloques marcados con el número 2 son permiten separar byte por byte la cadena de 5 bytes recibida para posteriormente procesarlos.

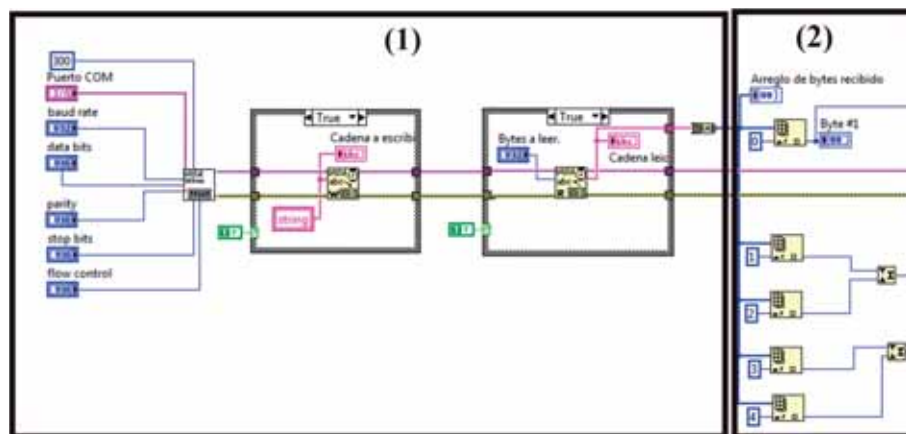


Figura 5.4: Diagrama de bloques para el esquema de radio frecuencia en su modo configuración.

5.2.2. Modo manual de la interfaz mediante el esquema de radiofrecuencia

A continuación se describe la interfaz gráfica en su modo manual a partir de la numeración propuesta en la Figura 5.5.

1. Esta sección permite monitorear de manera gráfica el nivel de líquido presente en cada uno de los tanques del sistema.
2. Este apartado muestra de manera numérica el caudal de entrada y la altura de cada uno de los tanques del sistema.

3. Esta sección permite controlar el encendido de la bomba (en caso de estar apaga el LED indicador de la sección 4 se encontrara apagada y en caso de estar encendida el LED indicador se prenderá) y el voltaje que el microcontrolador entregara a cada uno de las electroválvulas.

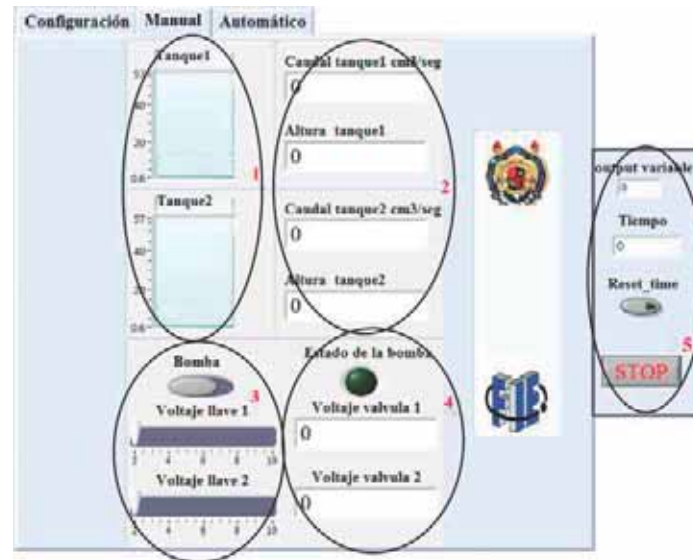


Figura 5.5: Aspecto de la segunda interfaz gráfica diseñada para el esquema de radio frecuencia en su modo manual.

4. Este apartado permite observar el estado de la bomba mediante un LED indicador y el voltaje que se le entrega a las electroválvulas mediante un indicador numérico para cada electroválvula.
5. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset_tiempo.

5.2.2.1. Diagrama de bloques del modo manual de la interfaz mediante el esquema de radiofrecuencia

En la Figura 5.6 se muestra el diagrama de bloques correspondiente al modo manual. Si llega un valor 0x38 se procesan los datos de entrada (número 1) correspondientes a los sensores y el valor que el usuario quiere en las electroválvulas y bomba se transmiten (número 2).

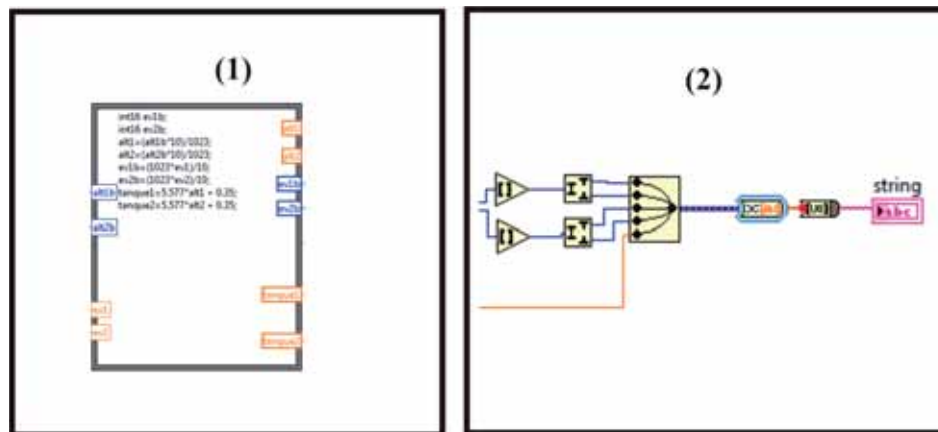


Figura 5.6: Diagrama de bloques para el esquema de radio frecuencia en su modo manual.

5.2.3. Modo automático de la interfaz mediante el esquema de radiofrecuencia

En la Figura 5.7 se muestra el diagrama de bloques correspondiente al modo automático.

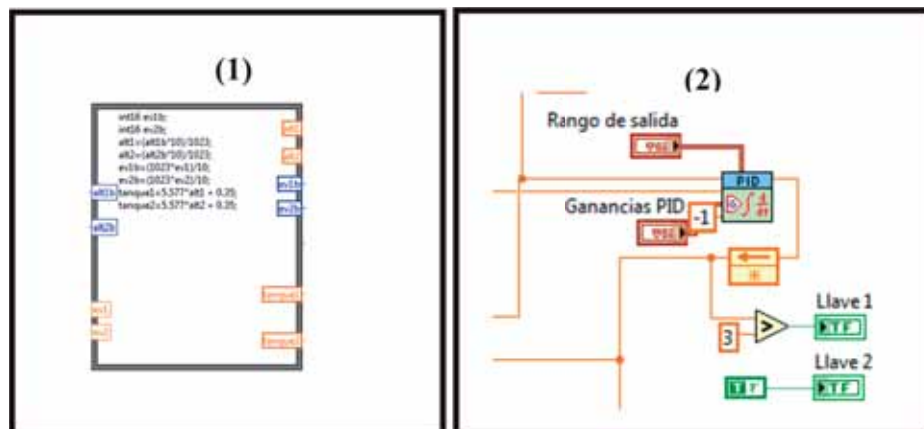


Figura 5.7: Diagrama de bloques para el esquema de radio frecuencia en su modo automático.

Si llega un valor 0x38 se procesan los datos de entrada (número 1) correspondientes a los sensores y se calcula mediante el controlador PID la acción de control de la electroválvula 1 mediante la librería PID de LabVIEW como se muestra en el número 2 (el control solo se realiza en la electroválvula 1, la electroválvula 2 permanece totalmente abierta y la bomba prendida durante el modo automático).

5.2.4. Modo automático para el tanque 1 de la interfaz mediante el esquema de radiofrecuencia

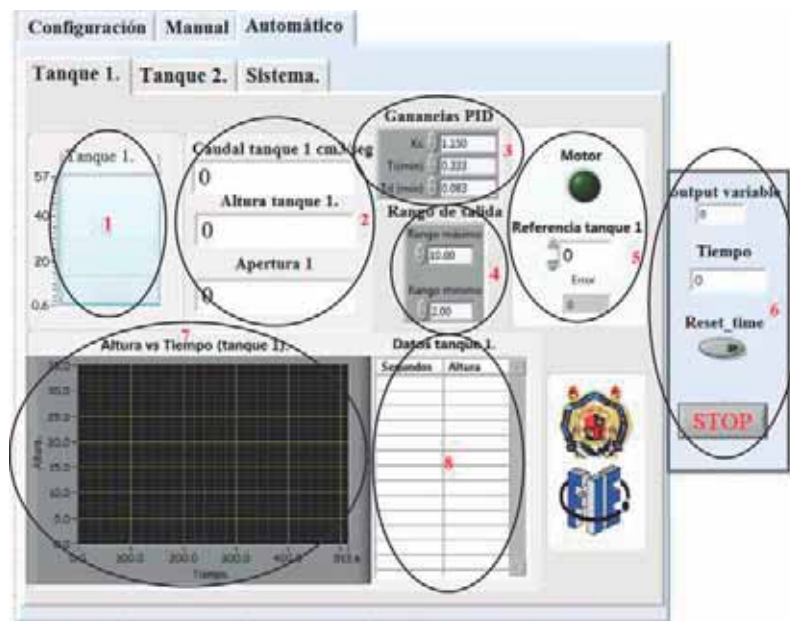


Figura 5.8: Aspecto de la segunda interfaz gráfica diseñada para el esquema de radio frecuencia en su modo automático, para el tanque 1.

A continuación se describe la interfaz gráfica en su modo automático para la el control del tanque 1 a partir de la numeración propuesta en la Figura 5.8.

1. Esta sección permite monitorear de manera gráfica el nivel de líquido presente en el tanque 1.
2. Este apartado muestra de manera numérica el caudal de entrada, la altura del tanque 1 y el voltaje aplicado por el controlador a la electroválvula 1.
3. Esta sección permite introducir las ganancias K_p , T_i , y T_d del controlador a utilizar en unidades de minutos (las ganancias pueden ser alteradas en cualquier momento antes y durante la ejecución del programa).
4. Este apartado permite determinar el rango máximo y mínimo de salida del controlador. Esta sección no necesita ser modificada ya que se en este caso el rango de voltaje de entrada a la electroválvula es de 2 a 10 volts.
5. Esta sección permite observar mediante un indicador tipo LED el estado del motor (bomba en funcionamiento LED encendido), también permite modificar la referencia

que va seguir el controlador y el error en voltaje que está viendo el controlador.

6. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset _tiempo.
7. Este indicador permite monitorear la evolución del control aplicado de manera gráfica, el eje Y esta expresado en centímetros y el eje X esta expresado en segundos.
8. Este indicador numérico va mostrando los datos de altura y tiempo que se presentan en la gráfica, estos valores son los que posteriormente van a ser guardados en el archivo determinado en el modo de configuración.

5.2.5. Modo automático para el tanque 2 de la interfaz mediante el esquema de radiofrecuencia

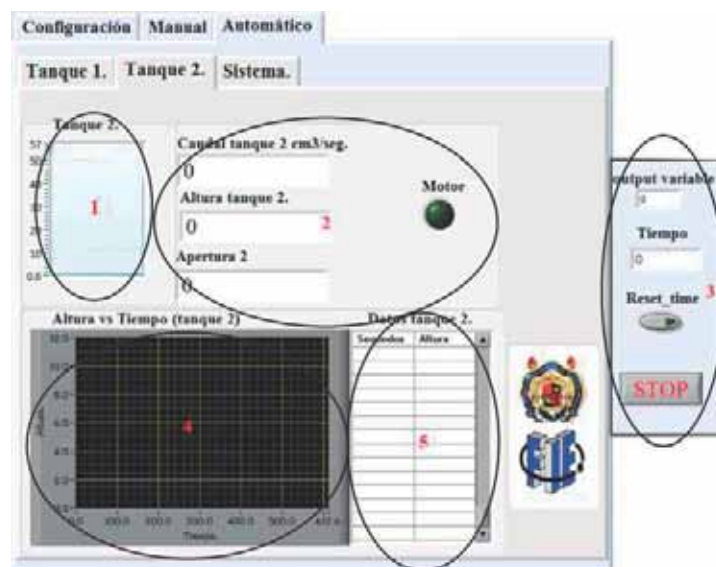


Figura 5.9: Aspecto de la segunda interfaz gráfica diseñada para el esquema de radio frecuencia en su modo automático para el tanque 2.

A continuación se describe la interfaz gráfica en su modo automático para la el monitoreo del tanque 2 a partir de la numeración propuesta en la Figura 5.9.

1. Esta sección permite monitorear de manera gráfica el nivel de líquido presente en el tanque 2.

2. Este apartado muestra de manera numérica el caudal de entrada, la altura del tanque 2 y el voltaje aplicado a la electroválvula 2 (para este caso el voltaje aplicado a la electroválvula 2 siempre es de 10volts), también nos permite observar mediante un indicador tipo LED el estado del motor (bomba en funcionamiento LED encendido).
3. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset_tiempo.
4. Este indicador permite monitorear la evolución de la altura del tanque 2 de manera gráfica, el eje Y esta expresado en centímetros y el eje X esta expresado en segundos.
5. Este indicador numérico va mostrando los datos de altura y tiempo que se presentan en la gráfica, estos valores son los que posteriormente van a ser guardados en el archivo determinado en el modo de configuración.

5.2.6. Modo automático para el sistema de la interfaz mediante el esquema de radiofrecuencia

A continuación se describe la interfaz gráfica en su modo automático para la visualización del sistema completo a partir de la numeración propuesta en la Figura 5.10.

1. Esta sección permite monitorear de manera numérica la altura del tanque 1 y 2, el voltaje entregado a las electroválvulas 1 y 2 y el estado de la bomba mediante un indicador tipo LED.
2. Esta sección permite monitorear de manera gráfica el nivel de líquido presente en los tanques 1 y 2.
3. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset_tiempo.

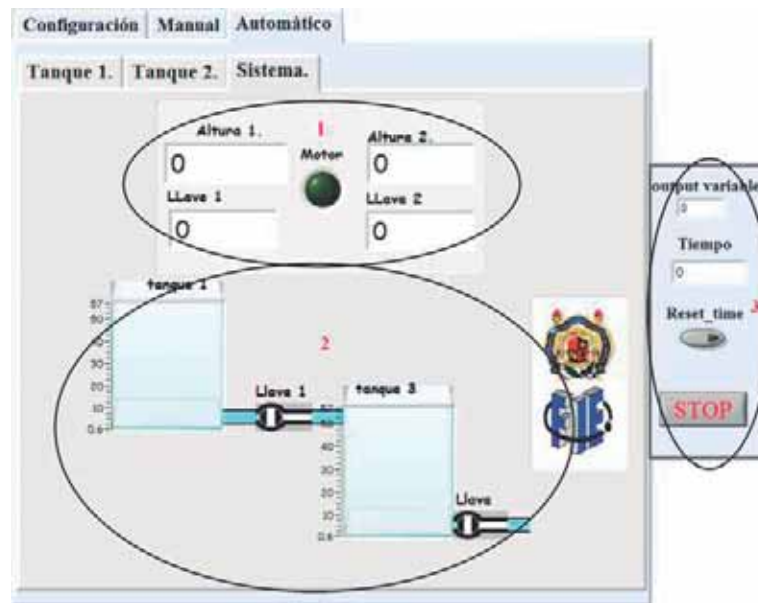


Figura 5.10: Aspecto de la segunda interfaz gráfica diseñada para el esquema de radio frecuencia en su modo automático, para el sistema completo.

5.3. Interfaz mediante los esquemas TCP/IP y TELNET

Se diseñaron dos interfaces gráficas, una que implementa el esquema TCP/IP y otra que maneja el esquema TELNET las cuales permiten controlar y monitorear de manera manual así como también poder monitorear el control PID del sistema (el control es aplicado solo al tanque 1), con la tarjeta basada en el microcontrolador PIC18F2550 (Figura 4.7) a través de Internet.

Tanto en el esquema de la Figura 5.11 y el de la Figura 5.12 se observa que el sistema consta de 2 computadoras interconectadas mediante el esquema que se elija ya sea TELNET o TCP/IP, la computadora marcada como servidor genera un puerto y espera que el cliente le solicite una conexión, la computadora marcada como cliente es el que tiene el control del sistema mediante la tarjeta basada en el microcontrolador PIC18F2550 (Figura. 4.7) y es el que solicita conexión al puerto de la computadora servidor mediante la dirección IP de la computadora servidor, una vez que la solicitud de conexión de la computadora cliente es aceptada por el servidor el sistema se encuentra conectado y la transferencia de datos es continua. La computadora cliente es la que tiene el control del sistema (la computadora servidor no tiene capacidad de modificar variables directamente

del sistema) esta puede determinar en qué momento el sistema se apaga, se prende, el voltaje de entrada a las electroválvulas y el momento en el que el sistema arranca el control PID.

En la Figura 5.8 se observa el esquema general seguido para la el monitoreo y control del sistema mediante el esquema TELNET.



Figura 5.11: Esquema para el monitoreo y control con el esquema TELNET.

En la Figura 5.12 se observa el esquema general seguido para la el monitoreo y control del sistema mediante el esquema TCP/IP.



Figura 5.12: Esquema para el monitoreo y control con el esquema TCP/IP.

En la Figura 5.13, Figura 5.14 y Figura 5.15 se observa el aspecto de la interfaz gráfica diseñada (tanto para el esquema TELNET como para el esquema TCP/IP) para la computadora que sirve como servidor. La interfaz gráfica diseñada consta de 3 apartados un apartado manual que permite el monitoreo del sistema, un apartado de configuración que permite seleccionar el puerto que de la tarjeta PIC18F2550 está usando para la recepción y transferencia de datos así como modificar las características de comunicación como la velocidad de transferencia de datos bytes de entrada entre otros del módulo a utilizar y un apartado de control el cual permite únicamente monitorear la evolución del controlador PID en el sistema.

Los diagramas de bloques seguidos para diseñar estas interfaces son los mismos seguidos por el esquema de radiofrecuencia (sección 5.2) los datos de control y monitoreo provenientes del microcontrolador son transmitidos y recibidos por el puerto serial para

que mediante LABVIEW se calculen las acciones de control correspondientes con la diferencia de que los datos de control de las electroválvulas, monitoreo de los sensores y estado de la bomba son transmitidos y recibidos mediante las librerías TCP/IP y TELNET (ya descritas en el capítulo 4) para observarlos en otra computadora.

5.3.1. Modo configuración de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora servidor

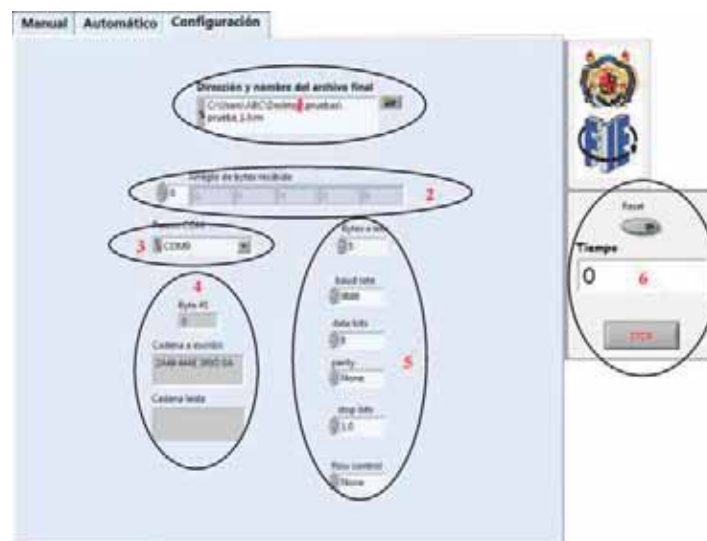


Figura 5.13: Aspecto de la interfaz gráfica para la computadora servidor en su modo configuración mediante TCP/IP o TELNET.

A continuación se describe la interfaz gráfica a partir de la numeración propuesta en la Figura 5.13.

1. Esta sección permite seleccionar la dirección y el nombre del archivo en donde se guardarán todos los datos de altura y tiempo de cada uno de los tanques del sistema. Este archivo es compatible con excel.
2. El microcontrolador transmite a la computadora un arreglo de 5 bytes con la información de los sensores de ultrasonido. Este apartado nos permite ver la cadena de datos entrante.
3. Este controlador permite seleccionar el puerto que el microcontrolador está usando para la recepción y transferencia de datos.

4. Esta sección permite observar en caso de recibir una cadena de datos, la cadena leída en forma hexadecimal, el byte numero 1 de la cadena leída el cual debe corresponder al hexadecimal 38, la cadena que se va a transmitir.
5. Esta parte permite configurar todo lo referente a las propiedades de comunicación con el módulo como la velocidad de transferencia de datos el bit de paridad, el bit de paro, el control de flujo, el número de bits de datos y el numero de bytes que se van a leer provenientes del microcontrolador en este caso son 5. No es necesario configurar esta sección ya que por default se encuentra configurada.
6. Este indicador permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en especifico se cuenta con el botón marcado como Reset_tiempo.

5.3.2. Modo manual de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora servidor

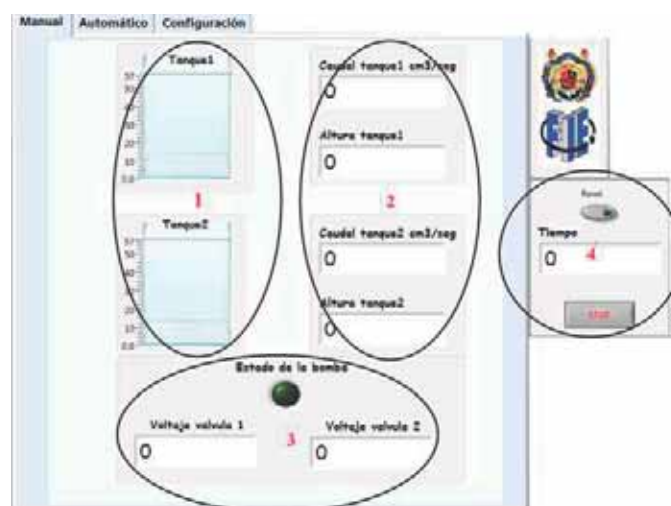


Figura 5.14: Aspecto de la interfaz gráfica para la computadora servidor en su modo manual mediante TCP/IP o TELNET.

A continuación se describe la interfaz gráfica a partir de la numeración propuesta en la Figura 5.14.

1. Esta sección permite monitorear de manera gráfica el nivel de líquido presente en cada uno de los tanques del sistema.

2. Este apartado muestra de manera numérica el caudal de entrada y la altura de cada uno de los tanques del sistema.
3. Este apartado permite observar el estado de la bomba mediante un LED indicador y el voltaje que se le entrega a las electroválvulas mediante un indicador numérico para cada electroválvula, el control de estos valores vienen dados de la computadora cliente que es la que controla de manera remota el sistema.
4. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset_tiempo.

5.3.3. Modo automático para el tanque 1 de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora servidor

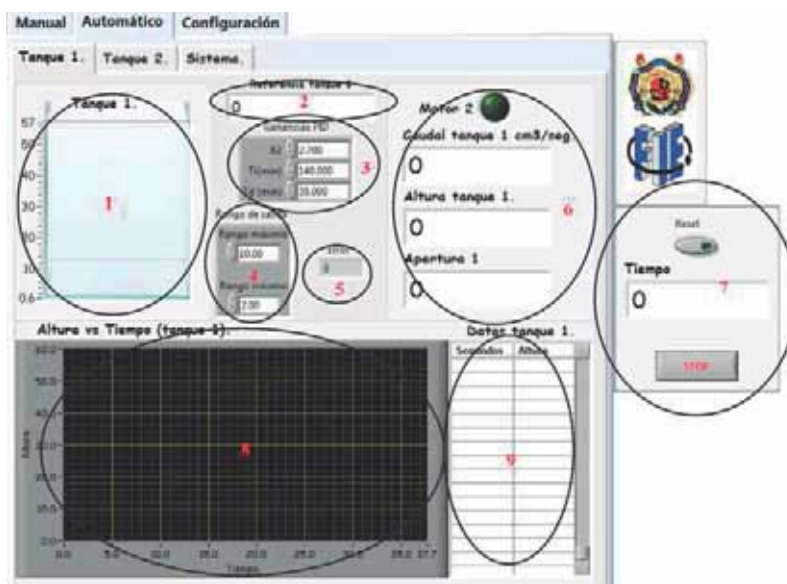


Figura 5.15: Aspecto de la interfaz gráfica para la computadora servidor en su modo automático para el tanque 1 mediante TCP/IP o TELNET..

A continuación se describe la interfaz gráfica a partir de la numeración propuesta en la Figura 5.15.

1. Esta sección permite monitorear de manera gráfica el nivel de líquido presente en el tanque 1.

2. Este apartado parte permite modificar la referencia que va seguir el controlador, es decir la altura a la que se quiere estabilizar el tanque.
3. Esta parte permite introducir las ganancias K_p , T_i , y T_d del controlador a utilizar en unidades de minutos (las ganancias pueden ser alteradas en cualquier momento antes y durante la ejecución del programa).
4. Esta sección permite determinar el rango máximo y mínimo de salida del controlador. Esta sección no necesita ser modificada ya que se en nuestro caso el rango de voltaje de entrada a la electroválvula la electroválvula es de 2 a 10 volts
5. Este indicador numérico permite observar el error en voltaje que está viendo el controlador.
6. Este apartado muestra de manera numérica el caudal de entrada, la altura del tanque 1, el voltaje aplicado por el controlador a la electroválvula 1 y observar mediante un indicador tipo LED el estado del motor (bomba en funcionamiento LED encendido).
7. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset _tiempo.
8. Este indicador permite monitorear la evolución del control aplicado de manera gráfica, el eje Y esta expresado en centímetros y el eje X esta expresado en segundos.
9. Este indicador numérico va mostrando los datos de altura y tiempo que se presentan en la gráfica, estos valores son los que posteriormente van a ser guardados en el archivo determinado en el modo de configuración.

5.3.4. Modo automático para el tanque 2 de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora servidor

A continuación se describe la interfaz gráfica a partir de la numeración propuesta en la Figura 5.13.

1. Esta sección permite monitorear de manera gráfica el nivel de líquido presente en el tanque 1.
2. Este apartado muestra de manera numérica el caudal de entrada, la altura del tanque 2 y el voltaje aplicado a la electroválvula 2 (para este caso el voltaje aplicado a la electroválvula 2 siempre es de 10volts), también nos permite observar mediante un indicador tipo LED el estado del motor (bomba en funcionamiento LED encendido).

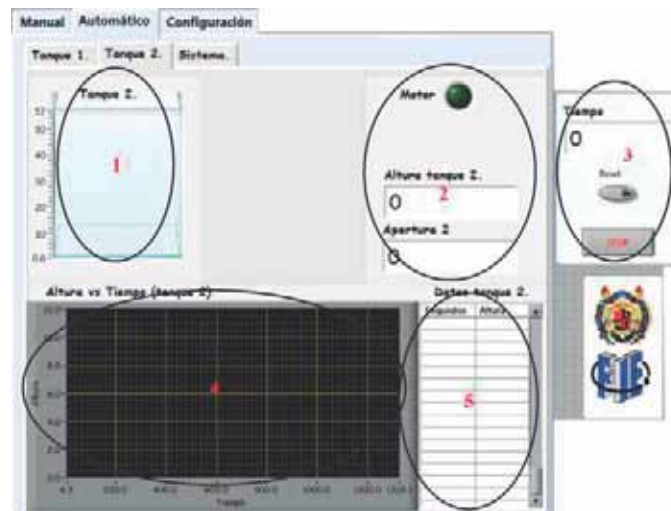


Figura 5.16: Aspecto de la interfaz gráfica para la computadora servidor en su modo automático para el tanque 2 mediante TCP/IP o TELNET.

3. Este indicador permite monitorear la evolución del control aplicado de manera gráfica, el eje Y esta expresado en centímetros y el eje X esta expresado en segundos.
4. Este indicador numérico muestra los datos de altura y tiempo que se presentan en la gráfica, estos valores son los que posteriormente van a ser guardados en el archivo determinado en el modo de configuración.
5. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset _tiempo.

5.3.5. Modo automático para el monitoreo del sistema de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora servidor

A continuación se describe la interfaz gráfica a partir de la numeración propuesta en la Figura 5.17.

1. Esta parte permite monitorear de manera gráfica el nivel de líquido presente en el tanque 1.
2. Esta sección permite monitorear de manera numérica la altura del tanque 1 y el voltaje entregado a la electroválvula 1.
3. Nos permite observar el estado de la bomba mediante un indicador tipo LED.
4. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset_tiempo.

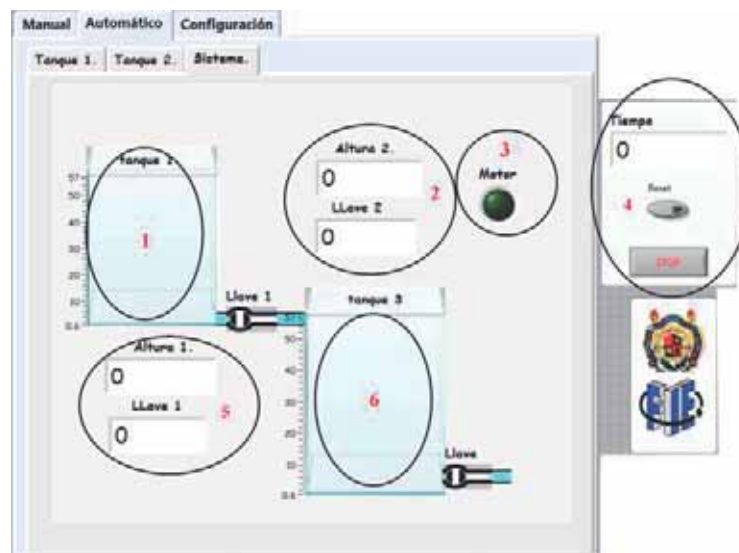


Figura 5.17: Aspecto de la interfaz gráfica para la computadora servidor en su modo automático para el sistema mediante TCP/IP o TELNET.

5. Esta parte permite monitorear de manera numérica la altura del tanque 2 y el voltaje entregado a la electroválvula 2.

6. Este indicador permite monitorear de manera gráfica el nivel de líquido presente en el tanque 2.

En la Figura 5.18, Figura 5.19 y Figura 5.20 se observa el aspecto de la interfaz gráfica diseñada (tanto para el esquema TELNET como para el esquema TCP/IP) para la computadora que sirve como cliente. La interfaz gráfica diseñada tiene el control total del sistema y consta de 2 apartados un apartado manual que nos permite el monitoreo y control manual del sistema y un apartado de control el cual nos permite determinar en qué momento arrancar el controlador PID para observar la evolución del controlador en el sistema.

5.3.6. Modo configuración de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora cliente

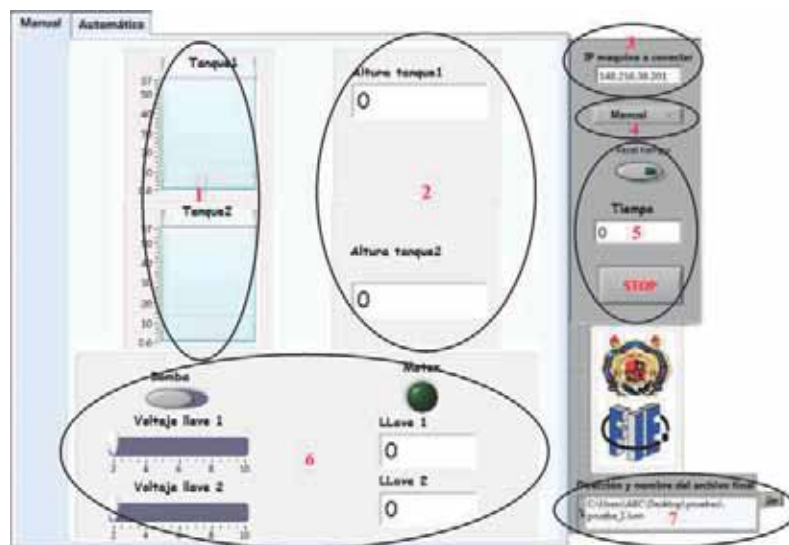


Figura 5.18: Aspecto de la interfaz gráfica para la computadora cliente en su modo manual mediante TCP/IP o TELNET.

A continuación se describe la interfaz gráfica a partir de la numeración propuesta en la Figura 5.18.

1. Este indicador permite monitorear de manera gráfica el nivel de líquido presente en cada uno de los tanques del sistema.
2. Este apartado muestra de manera numérica la altura de cada uno de los tanques del sistema.

3. Esta parte permite introducir la dirección IP de la computadora servidor con la que se puede establecer la comunicación, si la dirección es una dirección dinámica es decir que el módem asigna una dirección diferente cada vez que la máquina solicita una dirección IP el programa se podrá comunicar solamente con las computadoras dentro de una red local es decir las computadoras que estén conectadas a Internet bajo un mismo módem, en caso de que la dirección sea una dirección estática se podrá acceder desde cualquier computadora en cualquier lugar.
4. Esta sección permite seleccionar el modo en el que vamos a operar el sistema ya sea en modo manual o automático.
5. Este indicador permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset_tiempo.
6. Este botón permite controlar el encendido de la bomba (en caso de estar apagada el LED indicador de la sección 4 se encontrara apagada y en caso de estar encendida el LED indicador se prenderá) y el voltaje que el microcontrolador entregara a cada una de las electroválvulas, además nos permite observar el estado de la bomba mediante un LED indicador y el voltaje que se le entrega a las electroválvulas mediante un indicador numérico para cada electroválvula.

5.3.7. Modo automático para el tanque 1 de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora cliente

A continuación se describe la interfaz gráfica a partir de la numeración propuesta en la Figura 5.19.

1. Este indicador permite monitorear de manera gráfica el nivel de líquido presente en el tanque 1.
2. Esta parte permite modificar la referencia que va seguir el controlador, es decir la altura a la que se quiere estabilizar el tanque.

3. Este apartado muestra de manera numérica el caudal de entrada, la altura del tanque 1, el voltaje aplicado por el controlador a la electroválvula 1 y observar mediante un indicador tipo LED el estado del motor (bomba en funcionamiento LED encendido).

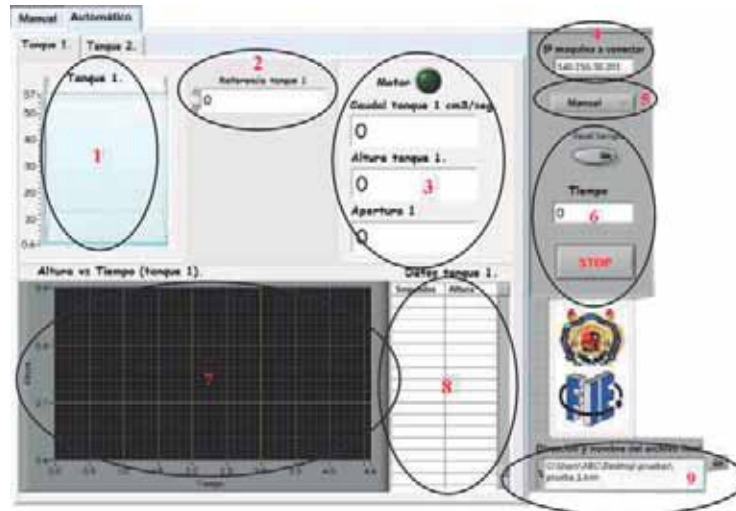


Figura 5.19: Aspecto de la interfaz gráfica para la computadora cliente en su modo automático para el tanque 1 mediante TCP/IP o TELNET.

4. Este indicador permite introducir la dirección IP de la computadora servidor con la que vamos a establecer la comunicación, si la dirección es una dirección dinámica es decir que el módem asigna una dirección diferente cada vez que la máquina solicita una dirección IP el programa se podrá comunicar solamente con las computadoras dentro de una red local es decir las computadoras que estén conectadas a Internet bajo un mismo módem, en caso de que la dirección sea una dirección estática se podrá acceder desde cualquier computadora en cualquier lugar.
5. Este apartado permite seleccionar el modo en el que vamos a operar el sistema ya sea en modo manual o automático.
6. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset_tiempo.
7. Este indicador permite monitorear la evolución del control aplicado de manera gráfica, el eje Y esta expresado en centímetros y el eje X esta expresado en segundos.

8. Este indicador numérico muestra los datos de altura y tiempo que se presentan en la gráfica, estos valores son los que posteriormente van a ser guardados en el archivo determinado en el modo de configuración.
9. Esta parte permite seleccionar la dirección y el nombre del archivo en donde se guardaran todos los datos de altura y tiempo de cada uno de los tanques del sistema. Este archivo es compatible con excel.

5.3.8. Modo automático para el tanque 2 de la interfaz mediante los esquemas TCP/IP y TELNET para la computadora cliente

A continuación se describe la interfaz gráfica a partir de la numeración propuesta en la Figura 5.20.

1. Esta sección permite monitorear de manera gráfica el nivel de líquido presente en el tanque 2.
2. Este apartado muestra de manera numérica el caudal de entrada, la altura del tanque 2, el voltaje aplicado por el controlador a la electroválvula 2 y observar mediante un indicador tipo LED el estado del motor (bomba en funcionamiento LED encendido).

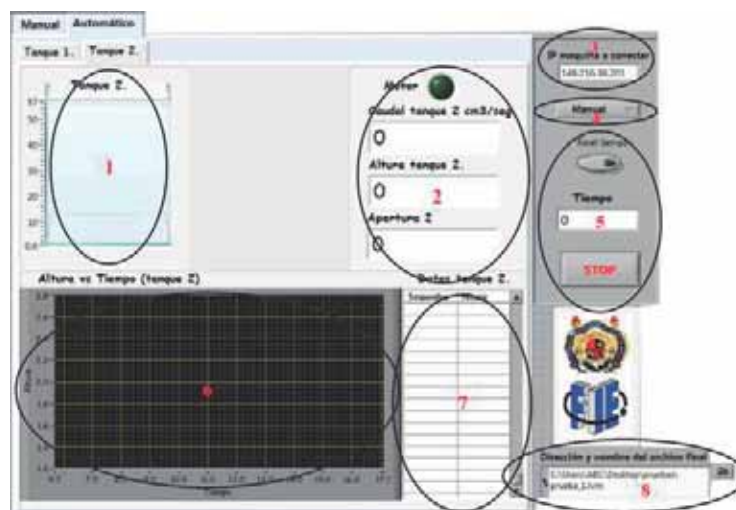


Figura 5.20: Aspecto de la interfaz gráfica para la computadora servidor en su modo automático para el tanque 2 mediante TCP/IP o TELNET.

3. Esta sección permite introducir la dirección IP de la computadora servidor con la que vamos a establecer la comunicación, si la dirección es una dirección dinámica es

decir que el módem asigna una dirección diferente cada vez que la máquina solicita una dirección IP el programa se podrá comunicar solamente con las computadoras dentro de una red local es decir las computadoras que estén conectadas a Internet bajo un mismo módem, en caso de que la dirección sea una dirección estática se podrá acceder desde cualquier computadora en cualquier lugar.

4. Esta sección permite seleccionar el modo en el que vamos a operar el sistema ya sea en modo manual o automático.
5. Esta sección permite observar el tiempo de ejecución del programa. Todos los datos de altura y tiempo son guardados de manera directa en el archivo que guarda el programa en caso de necesitar limpiar el archivo para guardar una prueba en específico se cuenta con el botón marcado como Reset_tiempo.
6. Este indicador permite monitorear la evolución del control aplicado de manera gráfica, el eje Y esta expresado en centímetros y el eje X esta expresado en segundos.
7. Este indicador numérico muestra los datos de altura y tiempo que se presentan en la gráfica, estos valores son los que posteriormente van a ser guardados en el archivo determinado en el modo de configuración.
8. Esta sección permite seleccionar la dirección y el nombre del archivo en donde se guardaran todos los datos de altura y tiempo de cada uno de los tanques del sistema. Este archivo es compatible con excel.

Capítulo 6

Pruebas y Resultados

Este capítulo muestra las pruebas realizadas con los 3 esquemas de comunicación que se aplicaron a lo largo del capítulo 4. Se muestran los esquemas de monitoreo y control que se siguieron. A lo largo del capítulo se muestran las respuestas obtenidas del sistema durante las diferentes pruebas del controlador PID implementado en lazo cerrado sobre el tanque 1 mediante los esquemas TCP/IP, TELNET, radiofrecuencia y las pruebas realizadas en lazo abierto mediante el esquema GSM.

6.1. Pruebas de la conversión de la señal PWM a una señal analógica

Como ya se menciona en la sección 3.2.2 se diseñó un circuito (Figura 6.1) para controlar las electroválvulas mediante la aplicación de un filtro pasa bajas a una señal PWM para obtener la señal de directa y la correspondiente amplificación con una ganancia de 2 de esta señal para manejar el rango apropiado de voltaje en las electroválvulas.

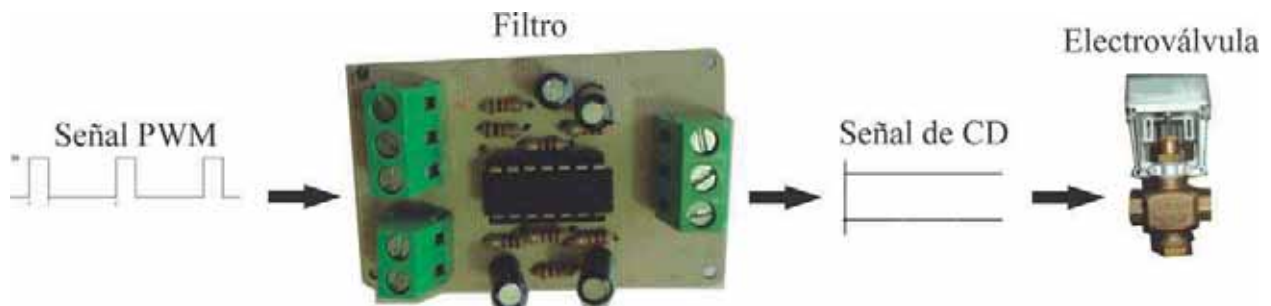


Figura 6.1: Esquema de funcionamiento del filtro pasa bajas diseñado.

Las pruebas que se presentan se realizaron variando el ciclo de trabajo de la señal PWM a la entrada del filtro y observando la forma en la que variaba la salida del filtro.

La primer prueba realizada fue introduciendo una señal PWM con un ciclo de trabajo del 10 % (Figura 6.2). A la salida del filtro observamos que la señal que se obtiene es una señal de 2 volts (la escala de las fotos es de 5volts/división) la cual se comporta como una señal de CD y ya no una señal de CD pulsante.

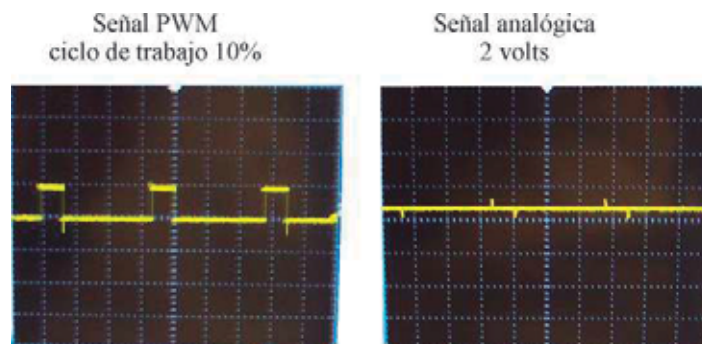


Figura 6.2: Prueba con una señal PWM a un ciclo de trabajo del 10 %.

La segunda prueba realizada fue introduciendo una señal PWM con un ciclo de trabajo del 50 % (Figura 6.3). A la salida del filtro observamos que la señal que se obtiene es una señal de 5 volts (la escala de las fotos es de 5volts/división) la cual se comporta como una señal de CD y ya no una señal de CD pulsante.

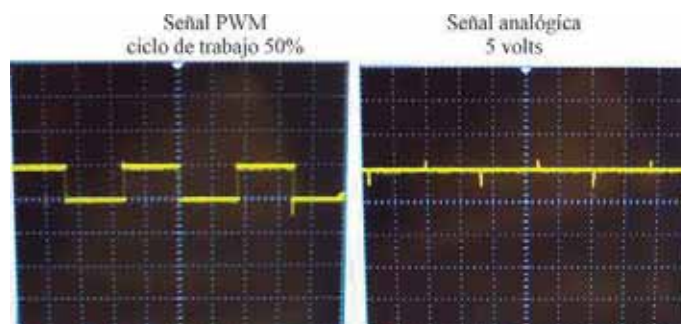


Figura 6.3: Prueba con una señal PWM a un ciclo de trabajo del 50 %.

La ultima prueba realizada sobre el circuito diseñado fue introduciendo una señal PWM con un ciclo de trabajo del 100 % (Figura 6.4). A la salida del filtro observamos que la señal que se obtiene es una señal de 10 volts (la escala de las fotos es de 5volts/división) la cual se comporta como una señal de CD y ya no una señal de CD pulsante.

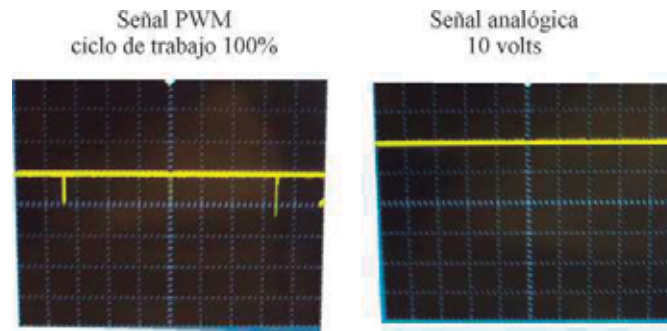


Figura 6.4: Prueba con una señal PWM a un ciclo de trabajo del 100 %.

De las pruebas realizadas se observa que las señales obtenidas son señales que pueden manejar directamente las electroválvulas que componen el sistema de nivel de líquido.

6.2. Monitoreo y control remoto mediante el protocolo ZigBee®

El periodo de muestreo del sistema para las pruebas mediante este esquema es de 500 milisegundos.

6.2.1. Pruebas de llenado y vaciado del tanque 1

Se realizaron 2 pruebas mediante esta interfaz gráfica, una prueba de llenado del tanque 1 (Figura 6.5) y una prueba de vaciado del tanque 1 (Figura 6.6).

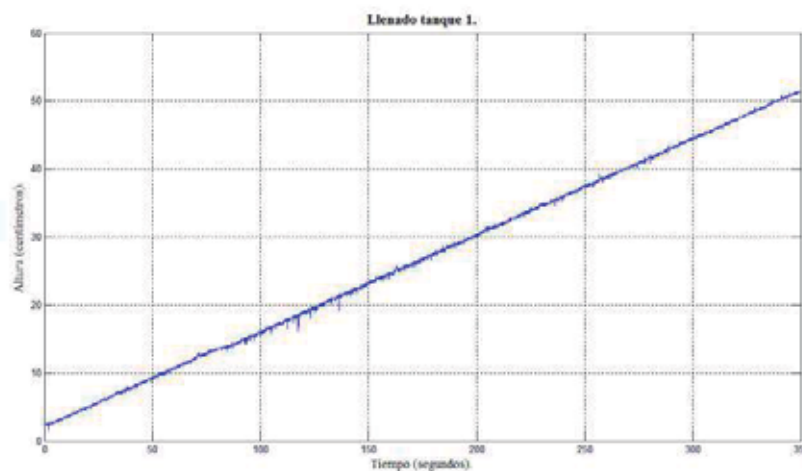


Figura 6.5: Prueba de llenado del tanque 1.

En la Figura 6.5 se observa el llenado lineal del sistema debido a que el caudal de entrada del tanque 1 es directamente la bomba de agua.

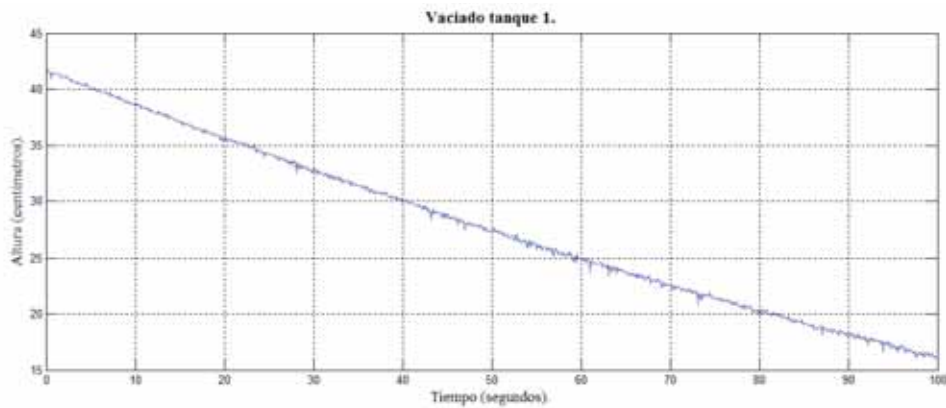


Figura 6.6: Prueba de vaciado del tanque 1.

En la Figura 6.6 se observa el vaciado del sistema, la respuesta tiende a hacer una curvatura al final debida a la presión ejercida en el agua para que salga en los últimos puntos es menor.

6.2.2. Prueba en lazo abierto del tanque 1 a una entrada escalón de 8 volts sobre la electroválvula 1

En la Figura 6.7 se observa la respuesta que tiene el sistema de primer orden ante una entrada de voltaje de 8 volts en la electroválvula 1.

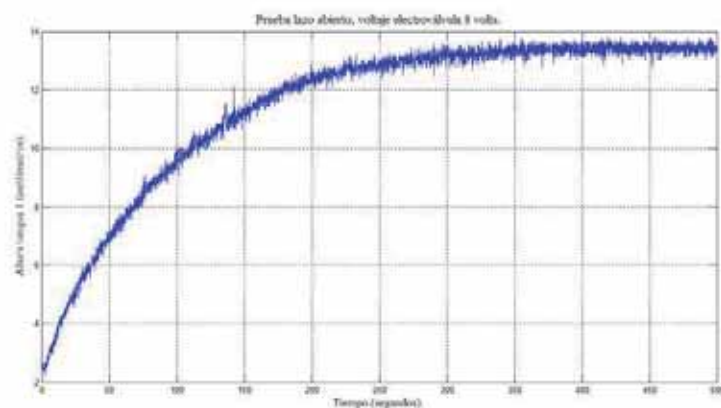


Figura 6.7: Prueba en lazo abierto ante una entrada escalón de 8 volts.

La estabilización ante esta entrada escalón de 8volts e obtiene a los 13.7cm.

6.2.3. Prueba en lazo abierto del tanque 1 a una entrada escalón de 5 volts sobre la electroválvula 1

En la Figura 6.8 se observa la respuesta que tiene el sistema de primer orden ante una entrada de voltaje de 5 volts en la electroválvula 1.

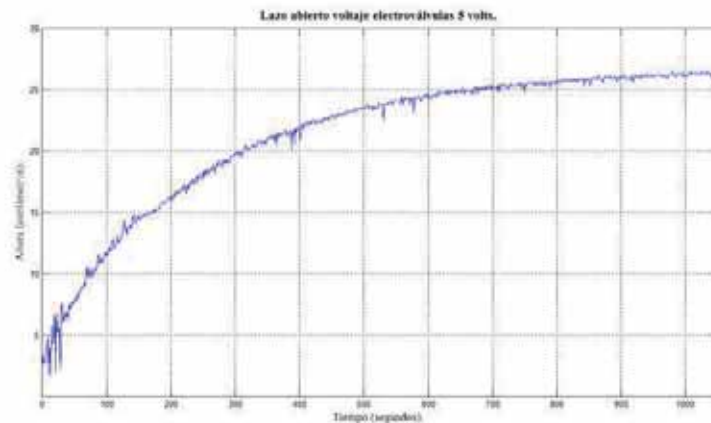


Figura 6.8: Prueba en lazo abierto ante una entrada escalón de 5 volts.

La estabilización ante esta entrada escalón de 8volts e obtiene a los 26.5cm.

6.2.4. Pruebas del controlador PID

Se realizaron varias pruebas mediante esta interfaz gráfica inalámbrica, la primer prueba se muestra en la Figura. 6.9 en donde se probó el controlador PID en base a las ganancias (propuestas en la sección 3.3.2) $K_p = 1.15, T_i = 20seg$ y $T_d = 5seg$.

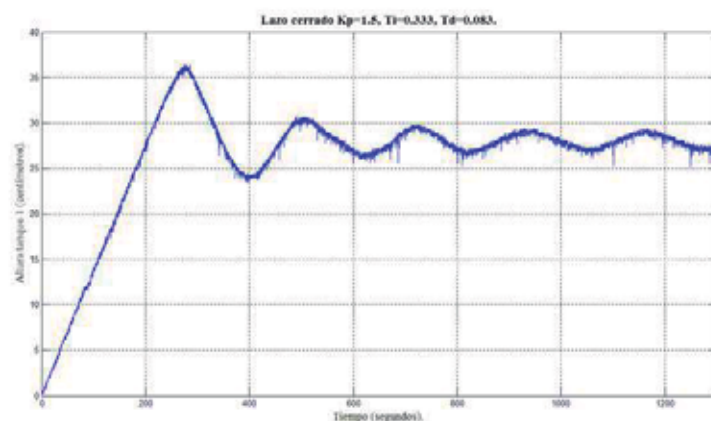


Figura 6.9: Prueba PID con módulo Xbee® $K_p = 1.15, T_i = .333min$ y $T_d = .0833min$.

La prueba corresponde a una referencia de 28 centímetros, se puede observar el sobre

impulso de 36.5cm y el sistema oscila alrededor del punto de referencia.

6.2.4.1. Prueba PID con una perturbación externa ganancias $K_p = 1.15, T_i = 20seg$ y $T_d = 5seg$.

El sistema se probó mediante las ganancias $K_p = 1.15, T_i = 20seg$ y $T_d = 5seg$ con una perturbación externa, esta perturbación correspondió a la entrada de un caudal distinto al de entrada al sistema. La respuesta del sistema se muestra en la Figura 6.10.

Aproximadamente a los 1380 segundos se introdujo una perturbación la cual alcanzó los 34 cm, el controlador responde de manera adecuada haciendo que la respuesta oscile al rededor de la referencia fijada de 28 centímetros.

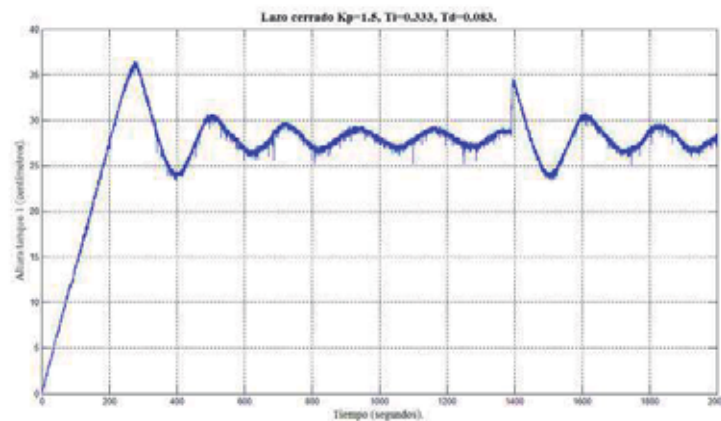


Figura 6.10: Prueba PID ante perturbación externa con módulo Xbee® $K_p = 1.15, T_i = .333min$ y $T_d = .0833min$.

6.2.4.2. Prueba PID con un cambio de referencia ganancias $K_p = 1.15, T_i = 20seg$ y $T_d = 5seg$.

El sistema se probó mediante las ganancias $K_p = 1.15, T_i = 20seg$ y $T_d = 5seg$ con un cambio de referencia del anteriormente fijado de 28cm a 18 cm. La respuesta del sistema se muestra en la Figura 6.11.

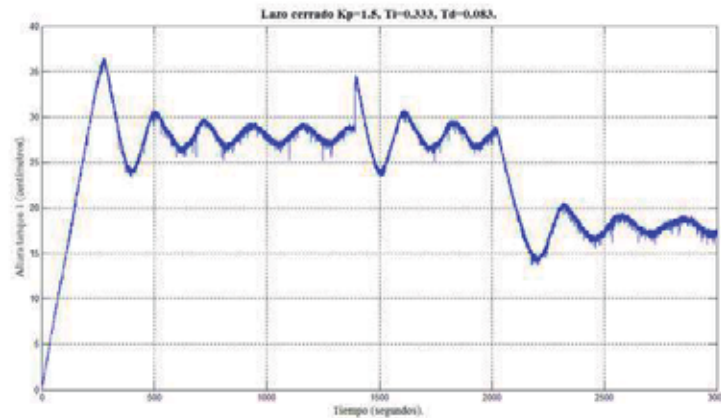


Figura 6.11: Prueba PID ante cambio de referencia con módulo Xbee® $K_p = 1.15$, $T_i = .333min$ y $T_d = .0833min$.

Aproximadamente a los 2050 segundos se introdujo el cambio de referencia del controlador, el controlador respondió de manera adecuada haciendo que la respuesta oscile al rededor de la referencia fijada de 18 centímetros.

A partir de esta respuesta se ajustan las ganancias de manera intuitiva, para mejorar la repuesta del sistema.

6.2.4.3. Prueba PID con ganancias $K_p = 5$, $T_i = .333min$ y $T_d = .083min$.

El sistema se probó mediante las ganancias $K_p = 5$, $T_i = .333min$ y $T_d = .083min$. La respuesta del sistema se muestra en la Figura 6.12.

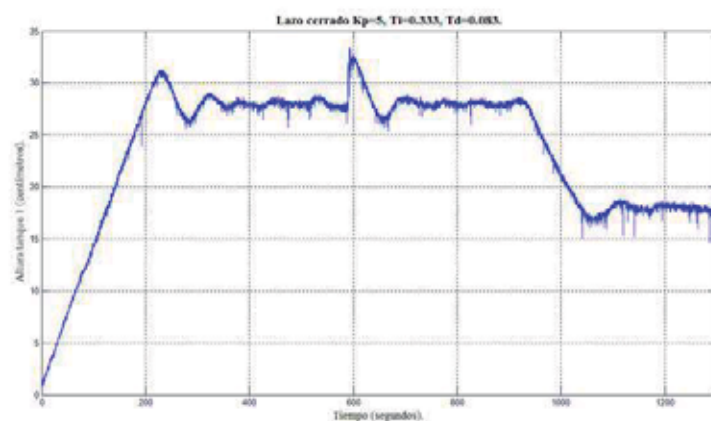


Figura 6.12: Prueba PID con módulo Xbee® $K_p = 5$, $T_i = .333min$ y $T_d = .0833min$.

La referencia fijada inicialmente es de 28cm, de la respuesta se observa que el sobrepulso es más pequeño en comparación a la respuesta del sistema que se observa

en la Figura 6.10, aproximadamente a los 380 segundos la respuesta del sistema tiende a estabilizarse. También se introdujo una perturbación aproximadamente a los 570 segundos la cual alcanzo los 33 cm, el controlador responde de manera adecuada haciendo que la respuesta del sistema regrese a la referencia de 28cm. Durante esta prueba aproximadamente a los 1080 segundos se realizó un cambio de referencia el cambio se hizo a 18cm, el controlador responde de manera adecuada haciendo que el sistema se estabilizó aproximadamente a los 18cm.

6.2.4.4. Prueba PID con ganancias $K_p = 8, T_i = .333min$ y $T_d = .083min$.

El sistema se probó mediante las ganancias $K_p = 8, T_i = .333min$ y $T_d = .083min$. La respuesta del sistema se muestra en la Figura 6.13.

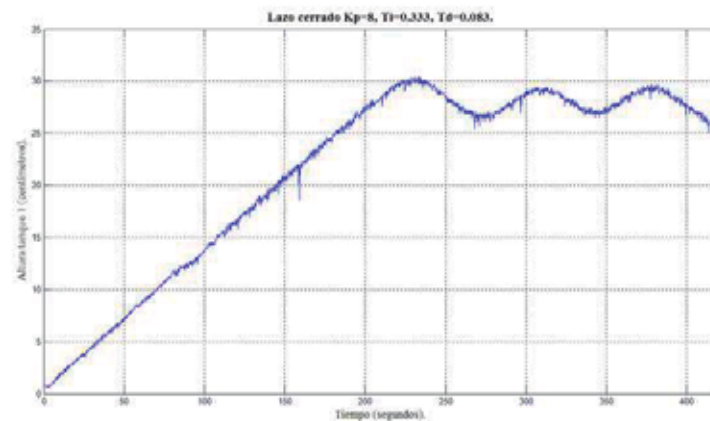


Figura 6.13: Prueba PID con módulo Xbee® $K_p = 8, T_i = .333min$ y $T_d = .0833min$.

La referencia fijada inicialmente es de 28cm, de la respuesta se observa que el sobreimpulso es más pequeño en comparación a la respuesta del sistema que se observa en la Figura 5.10, pero el sistema tiende a hacerse mas oscilatorio al rededor de la referencia del controlador.

6.2.4.5. Prueba PID con ganancias $K_p = 8, T_i = .3333min$ y $T_d = .9min$.

El sistema se probó mediante las ganancias $K_p = 8, T_i = .333min$ y $T_d = .083min$. La respuesta del sistema se muestra en la Figura 6.14.

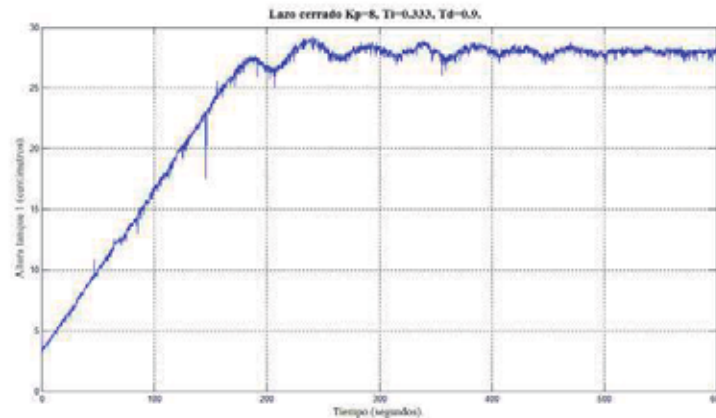


Figura 6.14: Prueba PID con módulo Xbee® $K_p = 8, T_i = .333min$ y $T_d = .9min$.

La referencia fijada inicialmente es de 28cm, de la respuesta se observa que el sobreimpulso inicial desaparece prácticamente y la estabilización del sistema se logra a los 300 segundos.

6.2.4.6. Comparación de las pruebas realizadas

En la Figura 6.16 se muestra la comparación de las diferentes pruebas realizadas al ajustar las ganancias del controlador PID.

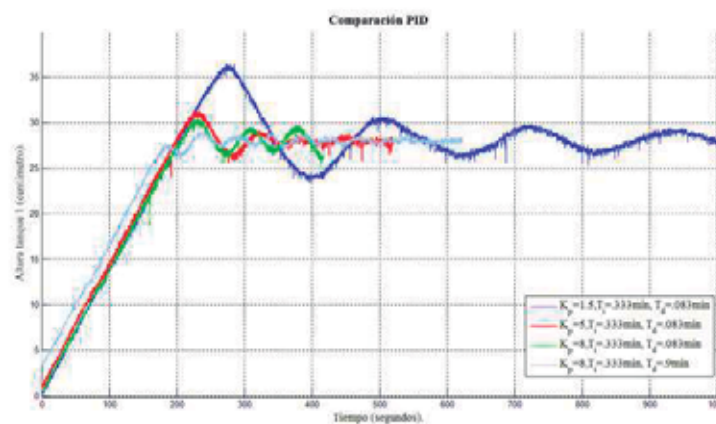


Figura 6.15: Comparación de las pruebas realizadas PID con módulo Xbee®.

6.2.4.7. Pruebas de distancia realizadas sobre los módulos Xbee-PRO®

Se realizaron 2 pruebas de distancia para probar el alcance de los módulos Xbee-PRO® en la primer prueba se logro transmitir y recibir datos a una distancia de 180 metros aproximadamente y en la segunda prueba se logro transmitir y recibir datos a una distancia de 150 metros aproximadamente como se muestra en la Figura 6.16.



Figura 6.16: Pruebas de distancia Xbee-PRO®.

6.2.4.8. Pruebas de distancia realizadas sobre los módulos Xbee®

Se realizaron 2 pruebas de distancia para probar el alcance de los módulos Xbee® en la primer prueba se logro transmitir y recibir datos a una distancia de 48 metros aproximadamente y en la segunda prueba se logro transmitir y recibir datos a una distancia de 50 metros aproximadamente como se muestra en la Figura 6.17.



Figura 6.17: Pruebas de distancia Xbee®.

6.3. Pruebas mediante el esquema TCP/IP

El periodo de muestreo del sistema para las pruebas mediante este esquema es de 500 milisegundos.

6.3.1. Prueba PID con ganancias $K_p = 9, T_i = .4min$ y $T_d = .4min$ mediante el esquema TCP/IP

El sistema se probó mediante las ganancias $K_p = 9, T_i = .4min$ y $T_d = .4min$. La respuesta del sistema se muestra en la Figura. 6.18.

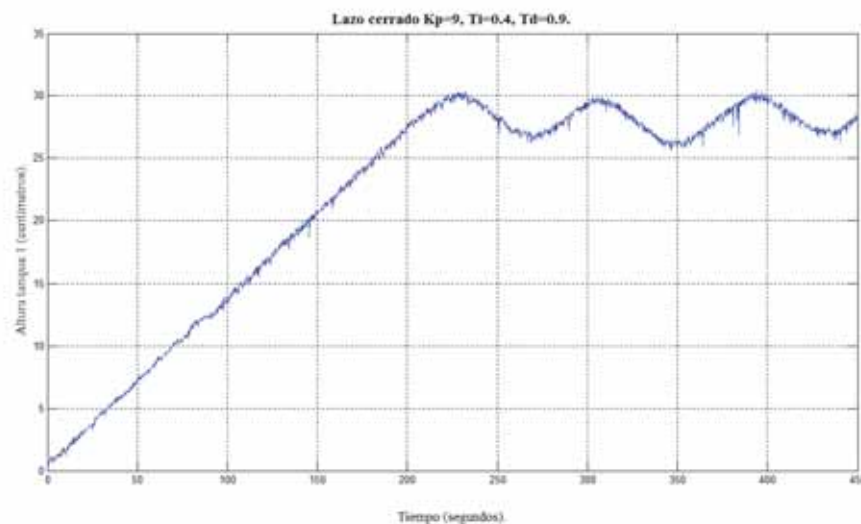


Figura 6.18: Prueba PID mediante esquema TCP/IP ® $K_p = 9, T_i = .4min$ y $T_d = .9min$.

La referencia fijada inicialmente es de 28cm, de la respuesta se observa que la respuesta del sistema se vuelve oscilatoria y las oscilaciones no tienden a disminuir. En la Figura 6.19 se muestra el aspecto real de la interfaz gráfica durante la realización de la prueba.

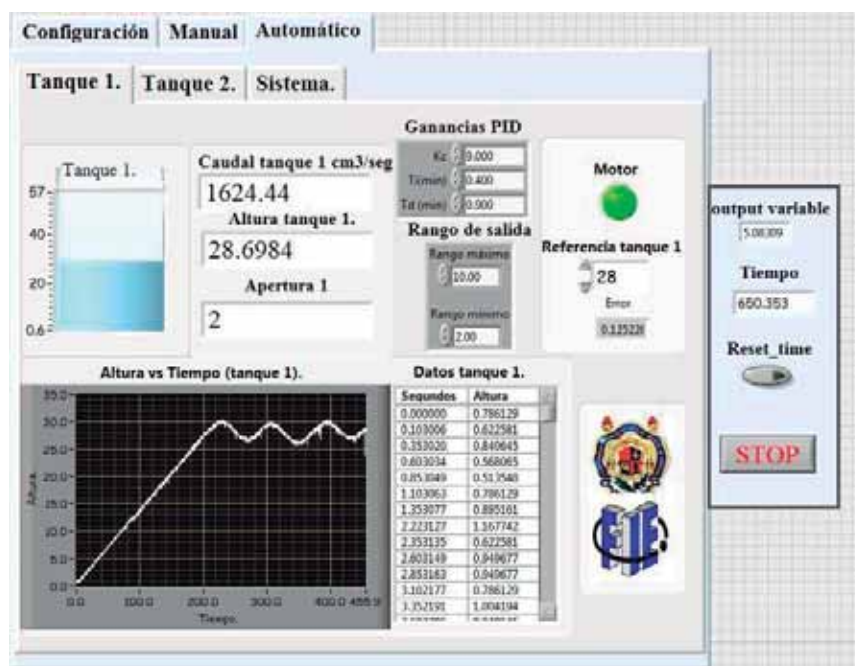


Figura 6.19: Prueba PID mediante esquema TCP/IP ® $K_p = 9, T_i = .4min$ y $T_d = .9min$.

6.4. Monitoreo remoto mediante el esquema TELNET

El periodo de muestreo del sistema para las pruebas mediante este esquema es de 500 milisegundos.

6.4.1. Prueba PID con ganancias $K_p = 8, T_i = .333min$ y $T_d = .9min$ mediante el esquema TELNET

El sistema se probó mediante las ganancias $K_p = 8, T_i = .333min$ y $T_d = .9min$ y una referencia de 28 cm. La respuesta del sistema se muestra en la Figura 6.20.

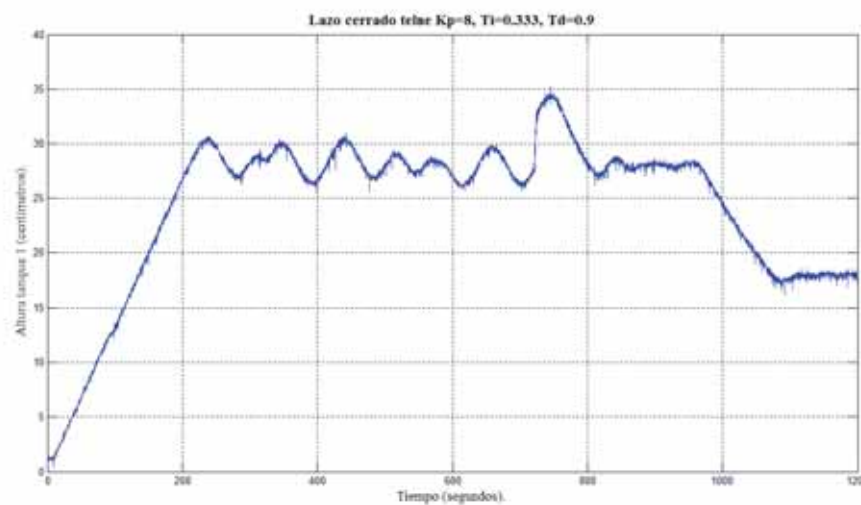


Figura 6.20: Prueba PID mediante esquema TELNET $K_p = 8, T_i = .333min$ y $T_d = .9min$.

La referencia fijada inicialmente es de 28cm, de la respuesta se observa que la respuesta del sistema tiende a estabilizarse a lo largo de la referencia, aproximadamente a los 700 segundos de la prueba se introdujo una perturbación externa en donde se introdujo agua y se llenó el tanque hasta los 34cm se observa que el controlador tiende a volver a ajustar la respuesta y mantiene la respuesta del sistema a 28cm. A los 1000 segundos se hizo un cambio de referencia a 18cm, se puede observar que el controlador tiende a seguir la referencia de manera adecuada casi sin oscilaciones y a los 1100 segundos el sistema se estabiliza. En la Figura 6.21 se muestra el aspecto real de la interfaz gráfica durante la realización de la prueba.

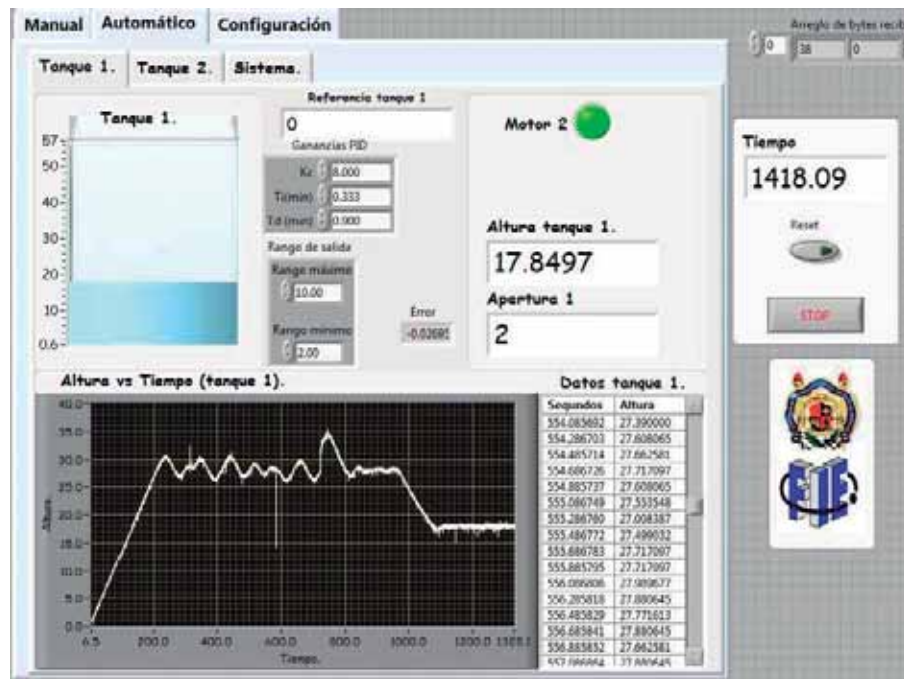


Figura 6.21: Prueba PID mediante esquema TELNET $K_p = 8, T_i = .333min$ y $T_d = .9min$.

6.4.2. Prueba PID con ganancias $K_p = 8, T_i = .333min$ y $T_d = .5min$ mediante el esquema TELNET

El sistema se probó mediante las ganancias $K_p = 8, T_i = .333min$ y $T_d = .5min$ y una referencia de 18 cm. La respuesta del sistema se muestra en la Figura 6.22.

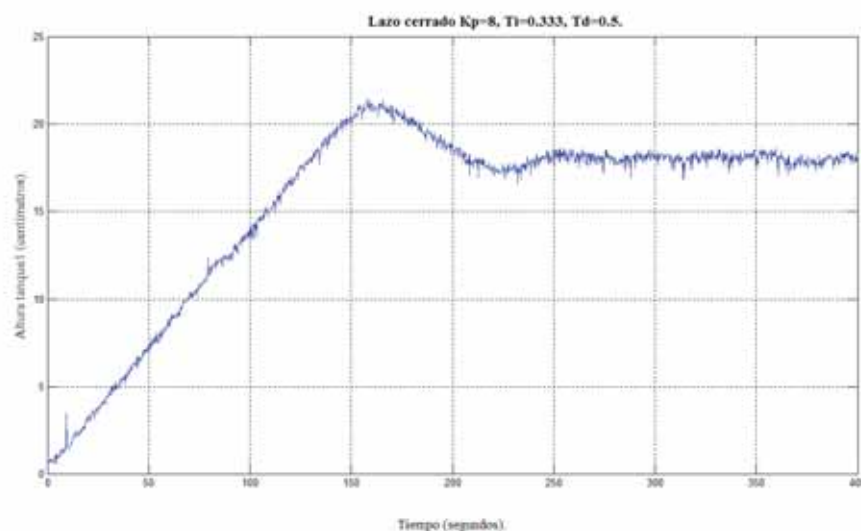


Figura 6.22: Prueba PID mediante esquema TELNET $K_p = 8, T_i = .333min$ y $T_d = .5min$.

La referencia fijada inicialmente es de 18cm, de la respuesta se observa que la respuesta del sistema tiende a estabilizarse a lo largo de la referencia prácticamente sin oscilaciones,

aproximadamente a los 250 segundos se logra la estabilización el sobre impulso que se presento es de 21 cm.

6.5. Monitoreo remoto mediante el esquema GSM

Se debe utilizar una fuente de 5V que pueda proporcionar de acuerdo al fabricante hasta 2 ampéres, para alimentar el módulo ya que el consumo de corriente puede variar dependiendo de la fuerza de recepción de red en el módulo. El envío de un SMS generalmente consume al rededor de 400mA. Cuando se usa el módulo para hacer una llamada el consumo de corriente es de alrededor de 420 mA. Estos 2 últimos parámetros se tienen en condiciones normales con una recepción de señal buena pero el consumo puede llegar a ser de hasta 2 ampéres.

6.5.1. Transmisión de llamadas

Se realizaron distintas pruebas con el módulo GSM una de ellas fue la transmisión de llamadas mediante el microcontrolador. El microcontrolador es programado como se explica en la sección 4.7.3.2 para transmitir llamadas a un número específico.



Figura 6.23: Prueba de transmisión de llamadas mediante el módulo GSM.

En la Figura 6.23 se muestra la prueba realizada, la cual consistió en que el microcontrolador configuró el módulo GSM y le especifica a que número de la red GSM se comunique.

6.5.2. Transmisión de mensajes SMS

Una vez que el sistema recibe el mensaje (#b0e00e00 explicado en la sección 4.7.5) para modificar los parámetros de las electroválvulas y la bomba el sistema regresa un mensaje con el estado de los tanques.

Se realizaron distintas pruebas con el módulo GSM una de ellas fue la transmisión de mensajes de texto mediante el microcontrolador. El microcontrolador es programado como se explica en la sección 4.7.3.3 para transmitir mensajes a un número específico con los datos del sistema.



Figura 6.24: Prueba 1 de transmisión de señales analógicas mediante mensajes de texto mediante el módulo GSM.

En la Figura 6.24 se muestra la primer prueba realizada, la cual consistió en que el microcontrolador adquirió 2 señales analógicas provenientes del sensor 1 y sensor 2 y transmitió los datos a un número específico de la red GSM, se adquirió una señal de 4.9 volts y una señal de 3.2 volts.



Figura 6.25: Prueba 2 de transmisión de señales analógicas mediante mensajes de texto mediante el módulo GSM.

En la Figura 6.25 se muestra la segunda prueba realizada, la cual consistió en que el microcontrolador adquirió 2 señales analógicas del sensor 1 y sensor 2 y transmitió los datos, se adquirió una señal de 0 v y una señal de 3.2 v.



Figura 6.26: Prueba 3 de transmisión de señales analógicas mediante mensajes de texto mediante el módulo GSM.

En la Figura 6.26 se muestra la tercera prueba realizada, la cual consistió en que el microcontrolador adquirió 2 señales analógicas del sensor 1 y sensor 2 y transmitió los datos, se adquirió una señal de 2.9 volts y una señal de 2.0 volts.

Posteriormente se adecuo la señal de voltaje a una señal de altura de para cada uno de los tanques.



Figura 6.27: Prueba 1 de transmisión de la altura de los tanques mediante mensajes de texto mediante el módulo GSM.

En la Figura 6.27 se muestra la primer prueba realizada, la cual consistió en que el microcontrolador adquirió las 2 señales analógicas provenientes de los sensores de ultrasonido procesara los datos y los modificara a unidades de altura y transmitiera los datos a un número específico de la red GSM, se adquirió una altura de 56.7cm para el tanque 1 y una altura de 38.1cm para el tanque 2.



Figura 6.28: Prueba 2 de transmisión de la altura de los tanques mediante mensajes de texto mediante el módulo GSM.

En la Figura 6.28 se muestra la segunda prueba realizada, la cual de igual forma consistió en que el microcontrolador adquirió las 2 señales analógicas provenientes de los sensores de ultrasonido procesara los datos y los modificara a unidades de altura y transmitiera los datos a un número específico de la red GSM, se adquirió una altura de 1cm para el tanque 1 y una altura de 38.1cm para el tanque 2.

Capítulo 7

Conclusiones

La tesis descrita a lo largo de los últimos capítulos presentó distintos esquemas y prototipos para el monitoreo y control de un sistema de nivel de líquidos de manera remota. El proyecto permite que el monitoreo y control se pueda llevar a cabo desde distintos esquemas de comunicación como los son Internet, radio frecuencia y GSM usando los prototipos y esquemas de conexión propuestos a lo largo del capítulo 4.

Los objetivos se alcanzaron satisfactoriamente, desde la instrumentación del sistema para la adquisición de las señales, el desarrollo de distintos tipos de tarjetas basadas en microcontroladores y el desarrollo de diversas interfaces gráficas que nos permitieron conectar el sistema desde una computadora remota y un celular para su control y monitoreo.

Se buscaron diferentes soluciones para la implementación de los esquemas de radiofrecuencia TCP/IP, TELNET y GSM para el caso de la radio frecuencia se encontró que los módulos Xbee® son muy versátiles en cuanto a las aplicaciones que se les puede dar, confiables para la transmisión de señales a largas distancias y accesibles ya que son muy fáciles de adquirir y su documentación es muy extensa. Para el caso de Internet se recurrió a librerías TCP/IP y TELNET de LabVIEW® las cuales nos permiten la transferencia de datos de manera remota desde Internet, su aplicación es directa ya que permite la transferencia de datos de manera sencilla siempre y cuando se cuiden los puertos para la transferencia de datos y la dirección IP entre las 2 computadoras interconectadas. En el caso del módulo GSM/GPRS SM5100B permitió realizar aplicaciones directamente utilizando la banda GSM o mediante mensajes de texto también conocidos como SMS.

Monitorear remotamente proporciona muchos beneficios, desde la reducción de trabajo al no tener que estar directamente en el sistema, aprender acerca de las tecnologías

disponibles hoy en día, tener informes del sistema en prácticamente tiempo real y enlazar distintos sistemas para su control.

Por el tipo de electroválvulas utilizadas el sistema de nivel de líquido se hace un sistema muy lento ya que cada una tarda en responder hasta 1 minuto entre apertura mínima y la apertura máxima, esto se pudo observar en las diferentes pruebas realizadas que llegaban a durar hasta 1500 segundos, las repuestas del sistema pueden ser mejoradas mediante el ajuste de las ganancias. Durante las pruebas se observan variaciones o pequeñas vibraciones en las mediciones de altura, las cuales son debidas a la sensibilidad de los sensores de ultrasonido que varían por el movimiento del agua dentro de los tanques.

Las pruebas con las interfaces realizadas consistieron en probar la respuesta del sistema de manera remota mediante un controlador PID dadas las ganancias calculadas mediante el método de Ziegler and Nichols. Durante estas pruebas se obtiene que la estabilización del sistema oscila dentro del punto que se determina como referencia por lo lento que es la respuesta del sistema debido al tipo electroválvulas utilizadas, posteriormente se modificaron las ganancias del sistema para mejorar la respuesta del sistema en base los conocimientos adquiridos en las materias de control analógico y digital y se logró mejorar el desempeño del sistema.

Los tiempos de respuesta para el caso de Internet y radiofrecuencia fueron de milisegundos, el periodo de muestreo del sistema para las pruebas mediante los diferentes esquemas fue de 500 milisegundos, en el caso del esquema GSM el tiempo de respuesta es más grande debido al tipo de servicio, por lo que este esquema se limitó a un control y monitoreo en lazo abierto. Hablar de ventajas de algún esquema respecto a otro es muy difícil ya que cada esquema de comunicación tiene sus aplicaciones definidas así como sus respectivos costos, sin embargo se probó que cualquiera de los 3 esquemas es idóneo para el monitoreo remoto de cualquier variable que se requiera independientemente del sistema que se desee monitorear.

Así como cada esquema tiene sus ventajas se tienen inconvenientes en el caso del módulo Xbee® la única limitante es la distancia de monitoreo ya que puede llegar a hacer muy corta, para el caso de las librerías utilizadas para Internet la limitante es que se ocupa de 2 computadoras conectadas a Internet para realizar el monitoreo y control lo cual para ciertas aplicaciones podría llegar a ser impráctico y para el caso del módulo GSM la única desventaja es el costo que se puede llegar a tener al realizar una aplicación ya

que los costos para transmitir mensajes o llamadas dependerán del proveedor del servicio GSM.

7.1. Trabajos futuros

Las líneas de investigación futuras así como el desarrollo de aplicaciones futuras es inmenso.

Como línea de investigación se podría seguir probando los modos de operación del módulo Xbee® ya que cuenta con más características como puertos de entrada y salida los cuales pueden ser utilizados para desarrollar aplicaciones enfocadas a sistemas más rápidos que requieran de monitoreos rápidos, en donde el módulo Xbee® sería muy útil por sus velocidades de operación.

En el caso de Internet buscar la solución a no depender directamente de 2 computadoras conectadas a la red y tratar de realizar lo que se hizo en esta tesis pero mediante 2 microcontroladores conectados en Internet y no mediante dos computadoras o hacer uso de una página WEB para el monitoreo y control.

Como línea de investigación para el modulo GSM sería seguir probando los distintos modos de operación del módulo y tratar de realizar la transmisión de los datos vía GPRS lo cual es más barato y de mayor capacidad a la hora de los datos que se transmiten en comparación a la transmisión vía SMS la cual está limitada a 160 bytes por cada transmisión realizada.

En caso de ser necesario seguir con esta tesis, los módulos hechos son susceptibles de mejoras y cambios para hacerlos más eficientes, se podrían mejorar los prototipos realizados es decir tratar de realizar tarjetas con microcontroladores mas pequeños o en todo caso tarjetas de montaje superficial.

Implementar algún otro esquema de comunicación que pueda sustituir a alguno implementado durante esta tesis que pueda hacerlo más eficiente o en todo caso ampliar las interfaces gráficas para tener más parámetros monitoreados y controlados en caso de ser necesario.

Implementación de una tarjeta única que contará con los 3 esquemas y no la implementación de los esquemas por separado como se realizó.

Apéndice A

LISTA DE SÍMBOLOS Y ABREVIATURAS

A	amperio.
ACK	ACKNOWLEDGEMENT, en comunicaciones, es un mensaje que se envía para confirmar que un mensaje o un conjunto de mensajes han llegado.
AM	Amplitud modulada.
API	Application Programming Interface, interfaz de programación de aplicaciones.
APF	All Pass Filter, Filtro Pasa Todo.
ASCII	American Standard Code for Information Interchange, Código Estadounidense Estándar para el Intercambio de Información.
AT	attention, instrucciones codificadas que conforman un lenguaje de comunicación entre el hombre y un terminal modem.
BPF	Band Pass Filter, Filtro Pasa Banda.
BSC	Base Station Controllers, controladores de estación base.
BSS	Base Station System, estación base del sistema.
BTS	Base Transceiver Stations, estaciones base transceptoras.
CA	corriente alterna.
CCIR	Comité Consultivo Internacional de Radio.
CD	corriente directa. Cm centímetros.
COFETEL	Comisión Federal de Telecomunicaciones en México.
DCS1800	Digital Cellular System 1800, variante de la norma GSM que utiliza la frecuencia de 1800 MHz.
DTE	Data Terminal Equipment, equipo terminal de datos.
DI	Data in, dato de entrada. DO Data out, dato de salida.
Endpoints	dispositivos finales.

EEPROM	Electrically-Erasable Programmable Read-Only Memory, tipo de memoria ROM que puede ser programado, borrado y reprogramado eléctricamente.
EGSM900	Extended GSM-900 Band, variante de la norma GSM que utiliza la frecuencia de 900 MHz.
EHF	Extremely High Frequency, 30-300 GHz.
EHz	ExaHertz, 10 ¹⁸ Hz.
ELF	Extremely Low Frequency, 3-30 Hz.
ETSI	European Telecommunications Standards Institute, Instituto Europeo de Estándares de Telecomunicaciones.
FDX	Full Duplex.
F/FDX	Full/Full Duplex.
FM	frecuencia modulada.
GND	GROUND , suelo o la tierra puede ser el punto de referencia en un circuito eléctrico.
GPRS	General Packet Radio Service, servicio general de paquetes vía radio extensión del GSM.
GPS	Global Positioning System, sistema de posición global GSM sistema global para las comunicaciones móviles.
GSM850	variante de la norma GSM que utiliza la frecuencia de 850 MHz.
HDX	Half Duplex.
HF	High Frequency, 3–30 MHz.
HLR	Home Location Register, Registro de Localización Local.
HP	Horse Power, caballo de potencia, 745.69987 Watts.
HPF	High Pass Filter, Filtro Pasa Altos.

Hz	hertz (símbolo Hz), unidad de frecuencia.
ICD	In-Circuit Debugger, programador para microcontroladores PIC fabricados por Microchip Technology.
IEEE802.15.4	estándar de redes inalámbricas de área personal.
Internet	International Network of Computers, red internacional de computadoras.
IP	Internet Protocol, protocolo de Internet.
K	ganancia estática.
Kb	Kilobyte, unidad de almacenamiento.
Kbps	kilobyte por segundo.
KHz	Kilohertz 1000Hz Kohms resistencia eléctrica, 1000ohms.
Kp	ganancia proporcional.
L	es la constante de tiempo muerto.
LabVIEW	LabVIEW (Laboratory Virtual Instrument Engineering Workbench), lenguaje de programación gráfico para el diseño de sistemas.
LED	Light-Emitting Diode, diodo emisor de luz.
LF	Low Frequency, 30–300 KHz.
LPF	Low Pass Filter, Filtro Pasa Bajos.
mA	miliampéres.
MAP	Mobile Application Part, Parte de Aplicación Móvil Mb megabyte.
Mbit/s	megabyte por segundo.
ME	Mobile Equipment, equipo móvil.
MF	Medium Frequency, 300–3000 KHz.
MHz	Megahertz, 1,000,000 Hz.

MS	Mobile Station, Estación Móvil.
MSC	Mobile Switching Center.
NMT	sistema de telefonía móvil analógico de cobertura escandinava.
NSS	Network Sub-System, subsistema de red.
OSI	Open Systems Interconnec.
OSS	Operation Sub-System, subsistema de operación.
PAN	ID identificador de red.
PCS1900	variante de la norma GSM que utiliza la frecuencia de 1900 MHz.
PHz	PetaHertz, 10 ¹⁵ Hz.
PICKIT	programador para microcontroladores PIC de Microchip Technology.
PID	Proporcional, Integral y Derivativo.
PLC	Programmable Logic Controller, controladores lógicos programables.
PWM	Pulse Width Modulation, señales de modulación de ancho de pulso.
RAM	Random Access Memory, Memoria de Acceso Aleatorio.
RBF	Reject Band Filter, Filtro Rechazo de Banda.
FCC	Federal Communications Comision.
RPM	revoluciones por minuto, unidad de frecuencia usada frecuentemente para medir la velocidad angular.
RS232	Recommended Standard 232.
s	segundos.
SHF	Super High Frequency, 3-30 GHz.
SIM	Subscriber Identity Module, módulo de identidad del subscriptor.
SMSC	Short Message Service Center, Centro de Servicio de Mensaje Corto.

SME	Short Messaging Entities, Entidades de Mensajería Corta.
SMS	Short Message Service, servicio de mensajes cortos.
SX	Simplex.
T	constante de tiempo aparente.
TACS	Total Access Communication System, servicio de telefonía móvil analógico puesto en funcionamiento en el Reino Unido en 1985.
TCP	Transmission Control Protocol Protocolo de Control de Transmisión.
TELNET	TELEcommunication NETwork, red de telecomunicación.
THz	TeraHertz, 10 ¹² Hz.
Ti	tiempo integral.
Td	tiempo derivativo.
UART	Universal Asynchronous Receiver-Transmitter, transmisor asíncrono universal (comunicación RS232 a un nivel de voltaje de 3.3 volts).
UHF	Ultra High Frequency, 300–3000 MHz.
USART	Universal Serial Asynchronous Receiver-Transmitter transmisor serial asíncrono universal (comunicación RS232 a un nivel de voltaje de 5volts).
USB	Universal Serial Bus, bus universal en serie.
V	voltaje.
Vac	voltaje de corriente alterna.
Vcd	voltaje de corriente directa.
VF	Voice Frequency, 0.3-3 KHz.
VHF	Very High Frequency, 30–300 MHz.
VLf	Very Low Frequency, 3–30 KHz.
VLr	Visitor Location Register, Registro de Localización del Visitante.

W	Watt.
WPAN	Wireless Personal Area Network, redes inalámbricas de área personal.
WWW	World Wide Web, sistema de distribución de información a través de Internet.
ZigBee	conjunto de protocolos de alto nivel de comunicación inalámbrica para su utilización con radiodifusión digital.

Apéndice B

COPIA DEL ARTÍCULO PUBLICADO. S. RAMIREZ, I. LÁZARO, A. BARAJAS,
“INSTRUMENTACIÓN, MONITOREO Y CONTROL DE UN SISTEMA DE NIVEL DE
LÍQUIDO VÍA INALÁMBRICA MEDIANTE LA TECNOLOGÍA ZIGBEE.” I CONGRESO
INTERNACIONAL DE INGENIERÍA ELECTRÓNICA, INSTRUMENTACIÓN Y
COMPUTACIÓN. MINATITLÁN VERACRUZ, MÉXICO



Instrumentación, Monitoreo y Control de un Sistema de Nivel de Líquido Vía Inalámbrica Mediante la Tecnología ZigBee.

S. Ramirez, *Member, IEEE*, I. Lázaro, *Member, IEEE*, A. Barajas, *Student Member IEEE*.

Abstract— This article presents the implementation, monitoring and control of a liquid level system using ZigBee wireless transmission technology, as well as a user interface using a monitoring center created in LabView. The system consists of two interconnected tanks where the control flow between them is achieved through proportional valves. The liquid level monitoring of each tank is made via ultrasonic sensors. The level variable acquisition and valve control is performed by analog to digital converters and PWM signals respectively. Once the system variables are measured, the microcontroller generates a wireless network based on Zigbee protocol to monitor and control the system variables with a graphical interface developed in LabView.

Temas claves—Sistemas de nivel de líquido, radiofrecuencia, monitoreo remoto, Xbee, LabView.

I. INTRODUCCIÓN

En algunos procesos industriales los sistemas hidráulicos son una parte fundamental debido a que el control de nivel de líquido en depósitos y el flujo entre ellos es un problema básico es estos procesos ya que se requiere que los líquidos sean almacenados en tanques y luego bombeados a otro tanque. Muchas veces el líquido será procesado por tratamientos químicos en los tanques, pero siempre el nivel de fluido en los tanques debe ser controlado o monitoreado y el flujo entre ellos regulado. Las variables más importantes a controlar o monitorear en dichos sistemas son el nivel de líquido, temperatura, viscosidad, flujo, etc.

El control o monitoreo del nivel y el flujo en tanques son el centro de todos los sistemas de ingeniería para procesos químicos. Actualmente los sistemas hidráulicos son utilizados en diversos procesos como: industrias petroquímicas, industrias de fabricación del papel, e industrias de tratamiento de agua.

Aunado a lo anterior, en muchos procesos industriales es necesario extremar las medidas de seguridad por el tipo de sustancias que se manejan o bien por las altas temperaturas que se tiene en el proceso, esto provoca que el personal no puede estar físicamente para monitorear o controlar el sistema y por esto se necesita realizar el monitoreo en forma remota, para lo cual puede emplearse algún protocolo de comunicación tales como: internet, radiofrecuencia entre otros, esto permite disminuir riesgos o peligros latentes que se puedan presentar al realizar dicha actividad [1], [2].

Este trabajo presenta la instrumentación, monitoreo y control de apertura o cierre de las válvulas de un modelo a escala de un sistema de nivel de líquido el cual puede ser utilizado en aplicaciones del tipo didácticas o de investigación. El monitoreo y control del sistema se realiza usando el lenguaje de programación gráfico LabView, a través de técnicas de instrumentación inalámbrica en base al protocolo ZigBee

El módulo didáctico construido permitirá a los estudiantes de la Facultad de Ingeniería Electrónica, tener un sistema físico en donde aplicar los conocimientos adquiridos en las áreas de Control, obteniendo de manera remota los resultados de las acciones de control aplicadas.

II. DESCRIPCIÓN DEL SISTEMA

El Sistema Hidráulico construido se muestra en la Fig. 1, está compuesto por dos tanques para almacenamiento de fluidos de 38.6x38.6x58 cm, con aproximadamente 74 litros de capacidad, estos están alimentados por una bomba de agua de 127 Vca monofásica, y a la salida de cada tanque se coloca una electroválvula del tipo proporcional marca Honeyweel utilizadas en la industria.

El sistema hidráulico construido es de 2 entradas y 2 salidas, las entradas corresponden a voltajes aplicados a las electroválvulas y las salidas corresponden a los niveles del líquido de cada tanque.

III. INSTRUMENTACIÓN

A. Sensores.

Para determinar el nivel de líquido se utilizan sensores ultrasónicos de la marca Banner (Q45U), este tipo de sensores

A. Barajas estudiante de ingeniería en electrónica de la Universidad Michoacana de San Nicolás de Hidalgo (E-mail: abc24@ieee.org).

I. Lázaro labora en la Universidad Michoacana de San Nicolás de Hidalgo, Av. Fco. J. Mújica, S/N, Ciudad Universitaria, Col. Felicitas del Río. (E-mail: ilazaro@ieee-sco.org).

S. Ramirez labora en la Facultad de Ingeniería Eléctrica de la Universidad Michoacana de San Nicolás de Hidalgo, Av. Fco. J. Mújica S/N, (email: szavala@zeus.umich.mx).

V. Torres, estudiante del Instituto Tecnológico de Morelia, Av. Tecnológico S/N Morelia Michoacán, (e-mail: vtg_1982@hotmail.com)



son programables en un rango de sensibilidad dentro de (25 cm. - 4 m.) como se observa en la Fig. 2.

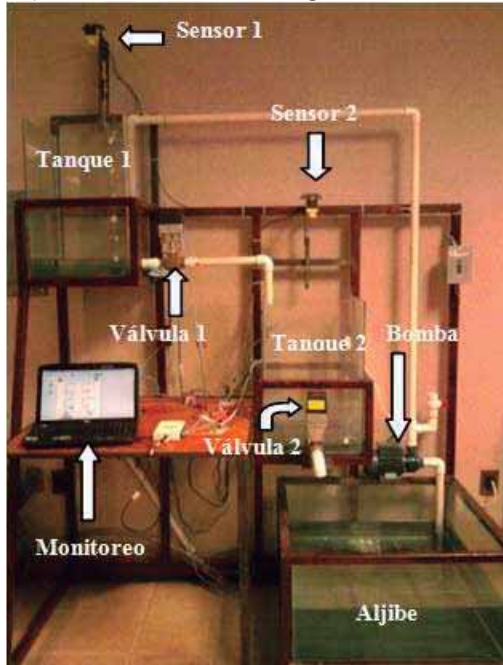


Fig. 1. Sistema de nivel de líquido.

Para el sistema descrito se utilizó un rango alrededor de 90 cm, esto permite sensar el nivel de líquido de los tanques desde 0 cm a 60 cm de altura cada uno. La Fig. 3 muestra el diagrama esquemático del sensor y su aspecto físico.

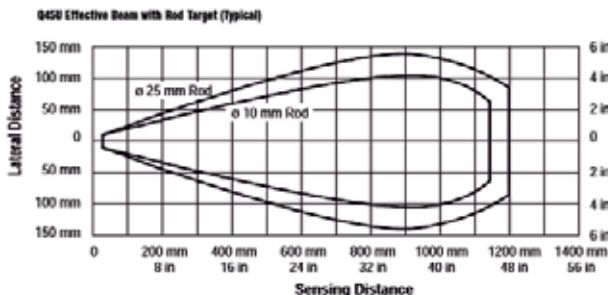


Fig. 2. Rango de operación de los sensores ultrasónicos.

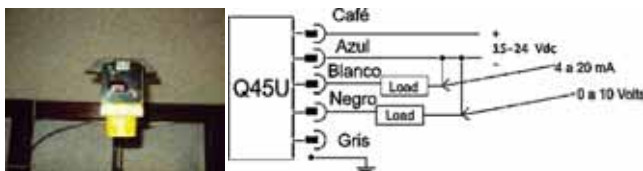


Fig. 3. Aspecto físico y esquemático del sensor Banner (Q45U).

La salida del sensor entrega un voltaje de variación lineal en el rango 0 a 10 Volts de CD. Para poder manejar salidas de voltaje en la red inalámbrica, se tienen que ajustar ya que se cuenta con 2 convertidores analógicos a digitales a 10 bits que manejan rangos de 0 a 5 volts provenientes de un

microcontrolador (PIC18f4550), por lo que las salidas de los sensores se ajustan a los voltajes de 0 a 5 volts mediante un arreglo de amplificadores operacionales (con una ganancia de $\Delta=0.5$) como se observa en la Fig. 4, y así lograr monitorear cada una de las alturas en tiempo real (se tiene un arreglo por cada uno de los sensores de ultrasonido).

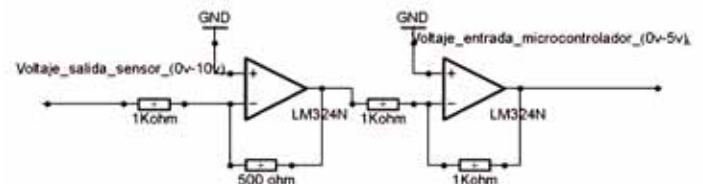


Fig. 4. Arreglo de operacionales para reducir el voltaje de los sensores 1 y 2 de ultrasonido.

B. Electroválvulas.

Para regular la salida de caudal o flujo de cada tanque que forma el sistema se utilizan dos electroválvulas de apertura y cierre proporcionales marca Honeywell, modelo ML6984 las cuales se muestran en la Fig. 5. Estas electroválvulas presentan un tiempo de apertura del 0% al 100% de aproximadamente de 60 seg.



Fig. 5. Electroválvula Honeywell, modelo ML6984.

Las electroválvulas trabajan tanto con entradas de voltaje (señales de 2 volts a 10 volts) o con entradas de corriente (señales de 4 mA a 20 mA). En este caso se usaron las entradas del tipo de voltaje, por lo que los voltajes aplicados a las electroválvulas permiten un modificar la apertura y cierre de las mismas proporcionalmente a la entrada de voltaje aplicado.

Las señales de voltaje se generan mediante dos señales de modulación de ancho de pulso (PWM) de 2.9 KHz con ciclo de trabajo variable de 10 bits proveniente del microcontrolador PIC16F84550 (una señal PWM por cada electroválvula). Como las señales PWM generadas son señales pulsantes de forma cuadrada estas contienen un conjunto de armónicos múltiplos de la frecuencia de la señal original y una componente de CD, para este caso ocupamos entregar la componente de CD y eliminar cualquier armónico presente en



las señales generadas. Como la componente de CD no contiene frecuencia basta con seleccionar un filtro del tipo pasa bajas a una frecuencia menor a la de las señales PWM generadas para eliminar los armónicos. Para este caso se realizó un filtro del tipo Butterworth de segundo orden con banda de paso plana críticamente amortiguada con una frecuencia de corte de 33 Hz el cual se muestra en la Fig. 6. Después de realizar el filtrado correspondiente, la señal CD varía de 0 a 5 volts con forme a como se modifique el ciclo de trabajo de cada una de las señales PWM, por lo que estas señales de CD se tienen que amplificar mediante un amplificador con una ganancia de 2 para ajustar las señales de 0 a 10 volts y así poder abarcar el rango completo de las electroválvulas de 2 a 10 volts. El circuito diseñado para esta etapa se muestra en la Fig. 6 (se tiene un arreglo por cada una de las electroválvulas).

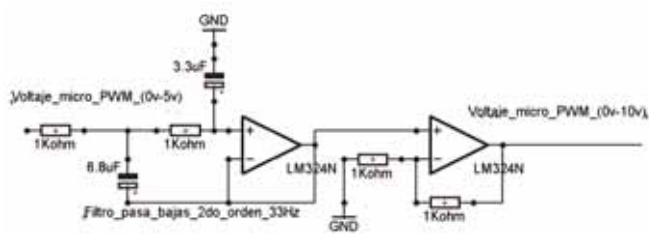


Fig. 6. Filtros pasa bajas de segundo orden a 33 Hz realizados para cada una de las señales PWM generadas.

C. Etapa de potencia.

La bomba consiste de un motor de inducción monofásico SIEMENS de 127 Vac de 1800 rpm con ½ HP de 5 amperes, por lo que la etapa de potencia para accionar y desactivar el motor se realizó directamente con un Triac (Q6010L5). El driver construido para controlar el encendido y apagado del motor tiene integrado un opto-acoplador (moc 3011), este se emplea para asilar la etapa de potencia y control y del Triac para el manejo de la carga. Esta etapa se muestra en la Fig. 7.

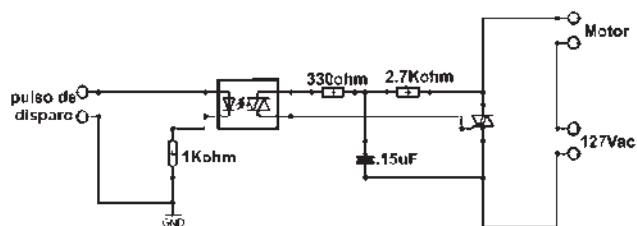


Fig. 7. Etapa de potencia para el majo de la bomba.

IV. TECNOLOGÍA ZIGBEE

La tecnología ZigBee tuvo su primera aparición en la industria en el año 2004, y su puesta a punto se dio hasta el año 2006. [3], [4], [5].

Al igual que Bluetooth, el origen del nombre ZigBee es un poco rebuscado, pero la idea se generó al observar lo que sucede en una colmena, en donde los insectos se comunican

mientras vuelan alrededor del panal. De acuerdo a los naturalistas una parte de la comunicación se produce a través del zumbido producido por las abejas [6].

Zigbee (zumbido de abejas) se define como el nombre de la especificación de un conjunto de protocolos de alto nivel de comunicación inalámbrica para su utilización con radios digitales de bajo consumo, está basada en el estándar IEEE 802.15.4, acceso físico y capa de acceso a los medios de comunicación (Physical Layer (PHY) y Media Access Layer (MAC) de redes inalámbricas de área personal (wireless personal area network, WPAN) [7]. Su objetivo son las aplicaciones que requieren comunicaciones seguras con baja tasa de envío de datos y maximización de la vida útil de sus baterías.

Como la tecnología ZigBee fue diseñada para aplicaciones de bajo consumo, se enmarca con facilidad en los sistemas embebidos, mercado donde la versatilidad y confiabilidad son importantes, sin grandes requerimientos de ancho de banda. Es especialmente útil para redes de sensores en entornos industriales, médicos y, sobre todo, domóticos.

Las comunicaciones Zigbee se realizan en la banda libre de 2.4GHz [8] (hay que tener cuidado ya que comunicaciones como Bluetooth y Wi-Fi trabajan en esta misma banda libre). Normalmente puede escogerse un canal dentro de la banda libre de 2.4 GHz con un ancho de banda de 5 KHz de entre 16 posibles para evitar ruido de cualquier otro dispositivo que no trabaje bajo el mismo protocolo, en la Tabla 1 [9] se muestran los diferentes canales con sus anchos de banda, así como el comando necesario a enviar al Xbee para programar el ancho de banda seleccionado.

Tabla 1. Posibles canales con sus anchos de banda de trabajo

Frecuencia GHz					
Canal	Hexadecimal	F. Inferior	F. Central	F. Superior	Comando AT
11	0x0B	2.4025	2.4050	2.4075	ATCH0B
12	0x0C	2.4075	2.4100	2.4125	ATCH0C
13	0x0D	2.4125	2.4150	2.4175	ATCH0D
14	0x0E	2.4175	2.4200	2.4225	ATCH0E
15	0x0F	2.4225	2.4250	2.4275	ATCH0F
15	0x10	2.4225	2.4250	2.4275	ATCH10
17	0x11	2.4325	2.4350	2.4375	ATCH11
18	0x12	2.4375	2.4400	2.4425	ATCH12
19	0x13	2.4425	2.4450	2.4475	ATCH13
20	0x14	2.4475	2.4500	2.4525	ATCH14
21	0x15	2.4525	2.4550	2.4575	ATCH15
22	0x16	2.4575	2.4600	2.4625	ATCH16
23	0x17	2.4625	2.4650	2.4675	ATCH17
24	0x18	2.4675	2.4700	2.4725	ATCH18
25	0x19	2.4725	2.4750	2.4775	ATCH19
26	0x1A	2.4775	2.4800	2.4825	ATCH1A

Frecuencia base 2.405GHz.

El alcance depende de la potencia de transmisión del dispositivo y del tipo de antenas utilizadas (cerámicas, dipolos, etc.). La velocidad de transmisión de datos de una red Zigbee es de hasta 256 Kbps para Xbee's de la serie 1 y de hasta 1 Mbps para Xbee's de la serie 2 [10].



El uso del protocolo Zigbee va desde reemplazar un cable por una comunicación serial inalámbrica, hasta el desarrollo de configuraciones punto a punto, multipunto o redes complejas de sensores.

Las conexiones mínimas que necesita el módulo Xbee para poder ser utilizado se muestra en la Fig. 8. Posteriormente de esto, se debe configurar según la aplicación requerida por el usuario.

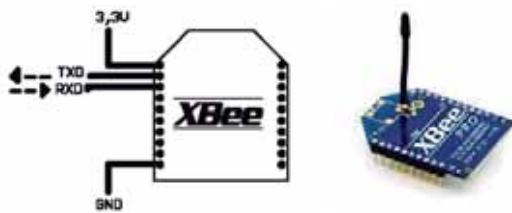


Fig. 8. Configuración básica Zigbee.

El módulo requiere una alimentación desde 2.8 a 3.4 V, la conexión a tierra y las líneas de transmisión de datos por medio del UART (TXD y RXD) para comunicarse con un microcontrolador, o directamente a un puerto serial para así poder ser leído o programado mediante comandos AT.

Existen 5 modos de operación de los módulos Xbee:

- **Modo de Conexión Transparente:** En este modo todo lo que ingresa por el pin 3 (Data in), es guardado en el buffer de entrada y luego transmitido y todo lo que ingresa como paquete RF, es guardado en el buffer de salida y luego enviado por el pin 2 (Data out).
- **Punto a Punto:** Es la conexión ideal para reemplazar comunicación serial por un cable inalámbrico.
- **Punto a Multipunto:** Permite transmitir información, desde la entrada serial de un módulo (DIN, pin 3) a uno o varios módulos conectados a la misma red de manera más controlada.
- **Cable Virtual I/O:** Se utilizan para crear un canal de comunicación de manera transparente entre los pines de un módulo y otro.
- **Conexión API:** Esta conexión, agrega información extra a los paquetes de datos RF. Ya no son enviados de forma transparente, sino que cada paquete de datos, son almacenados dentro de un conjunto de bytes con una estructura definida (frame).

Para este proyecto se utilizó un modo de conexión Punto a Multipunto el cual nos permite transmitir la información adquirida de los sensores a uno o varios módulos.

Una de las principales características de estos módulos Zigbee es que se pueden usar configurar, transmitir y recibir datos desde un puerto serial de una computadora personal o cualquier dispositivo que cuente con puerto serial como es el caso de los microcontroladores.

Así para esta aplicación se diseñó un módulo para conectar un microcontrolador PIC18F4550 directamente al módulo Zigbee como se muestra en el diagrama de bloques de la Fig. 9 y su fotografía del módulo físico en la Fig. 10, este módulo recibe los valores obtenidos por el microcontrolador PIC18F4550 que está conectado al sistema de nivel de líquido para obtener los valores de los dos sensores de ultrasonido (que indican el nivel de líquido) y los transmite en forma inalámbrica, además este módulo también es capaz de recibir del segundo módulo de Xbee el nivel de voltaje que se aplica a cada una de las electroválvulas y el estado de la bomba (prendido o apagado), pasando esta información al microcontrolador para interactuar sobre el sistema de nivel de líquido.

El segundo módulo diseñado tiene un Xbee y un convertidor (MAX232) para ajustar las señales provenientes del Xbee a puerto serial de una computadora personal donde se tiene la interfaz gráfica diseñada en un ambiente de programación de LabView para el manejo y control vía remota del sistema de nivel de líquido. Como se muestra en el diagrama de bloques y su fotografía en la Fig. 11.

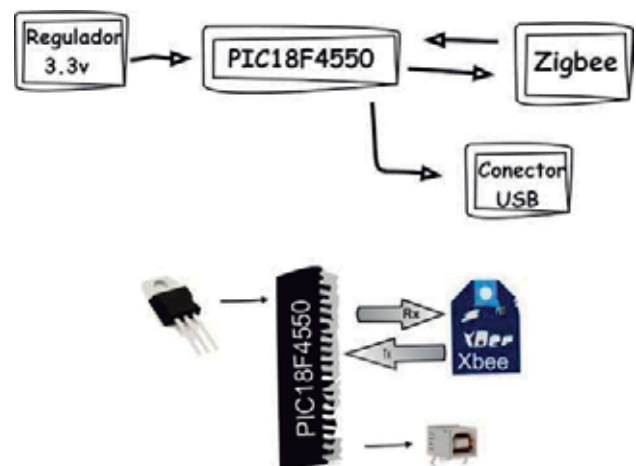


Fig. 9. Diagrama de bloques de la etapa de diseño del Módulo Zigbee USB.

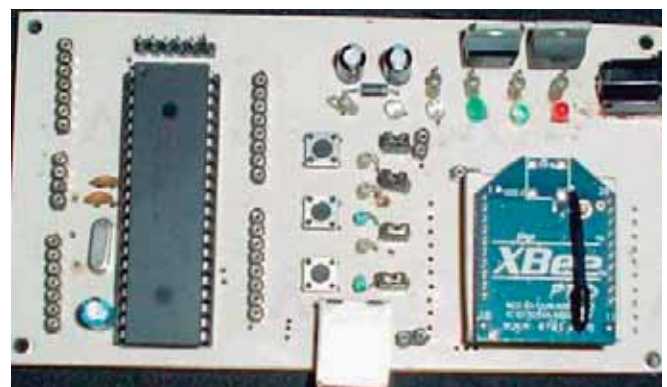


Fig. 10. Fotografía del aspecto físico del módulo Zigbee USB.



V. DESCRIPCIÓN DEL PROGRAMA DE VISUALIZACIÓN

Con el fin de monitorear las variables físicas del sistema, se desarrolló un programa o instrumento virtual (VI) usando la plataforma de LabView. El VI tiene la capacidad de observar en tiempo real el estado del nivel de líquido de cada uno de los tanques, el monitoreo y control de cada una de las electroválvulas del sistema y el encendido y apagado de la bomba.

La velocidad de respuesta de Zigbee es muy rápido para este caso el tiempo que se tiene desde que se adquieren los datos del sistema y se transmiten a la interfaz es de 15 mS. Este sistema es relativamente lento por lo que este tiempo de respuesta es muy adecuado cabe mencionar que la velocidad de transmisión puede ser modificada de acuerdo a la aplicación que se tenga y este tiempo se reduce conforme mas módulos Zigbee estén interconectados entre sí. En la Fig. 12 se muestra el panel frontal (interfaz de usuario) del instrumento virtual diseñado.

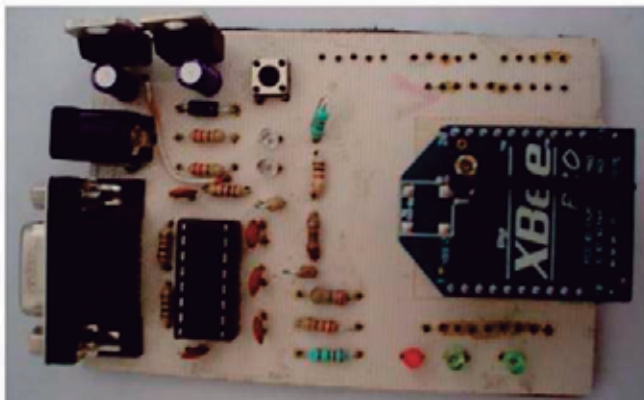
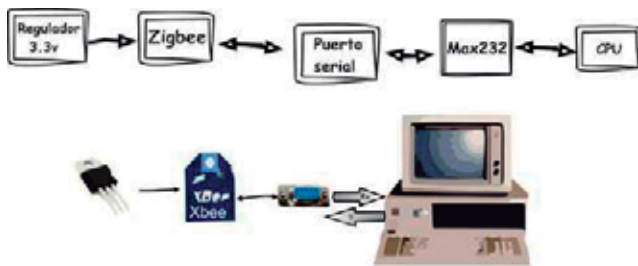


Fig. 11. Etapa de diseño y aspecto físico del módulo Xbee RS232.



Fig. 12. Vista general de la interfaz del sistema.

En el Panel Frontal se aprecian el monitoreo de las variables de caudal y altura de cada tanque, tanto de manera visual como numérica, en la parte inferior de la interfaz se tiene acceso al control de apertura de las electroválvulas y un botón de paro o encendido de la bomba (la operación del usuario sobre la bomba se visualiza a través de un led de encendido o apagado). La interface gráfica permite que el usuario tenga control de apertura o cierre de las electroválvulas. El usuario controla directamente el voltaje que se aplica a cada electroválvula (por lo que no se aplica ningún algoritmo de control directamente).

VI. PRUEBAS

En la Fig. 13 se muestra el diagrama de bloques que se implementa para comprobar la funcionalidad del sistema en conjunto.



Fig. 13. Diagrama de bloques sistema de nivel de líquido con la tecnología Zigbee.

En la Fig. 14 se muestra el funcionamiento de la interface grafica la cual nos muestra la altura de cada tanque después de haber mandado los comandos de encendido de la bomba (led en estado alto nos indica que la bomba esta prendida) y el valor de apertura de las electroválvulas (para esta prueba se indico un voltaje de 7.8 volts para la electroválvula 1 y 2 volts para la electroválvula 2). Una vez hechas las pruebas correspondientes la interface almacena los datos de altura obtenidos en un archivo compatible con el programa de Excel o Matlab para su análisis correspondiente.



Fig. 14. Monitoreo de los tanques 1 y 2.

VII. CONCLUSIONES.

Se ha presentado la instrumentación, monitoreo y control de un sistema de nivel de líquido vía remota por radiofrecuencia utilizando la tecnología de transmisión inalámbrica ZigBee de dos tanques interconectados, los sensores y actuadores empleados en el sistema por sus características industriales, permite tener un modelo a escala de sistemas típicos empleados en procesos industriales, tales como sistemas de embotellado, fermentación, producción de alimentos y plantas de tratamiento de aguas residuales, esto permite tener un proceso a escala en el cual en proyectos futuros se podrá probar diversos algoritmos de control, desde los clásicos PID's hasta técnicas modernas de control de manera directa.

La interfaz gráfica desarrollada permite el monitoreo remoto para observar el comportamiento del sistema desde un sitio remoto sin tener que estar presentes en el laboratorio donde está instalado el equipo.

Básicamente la idea sobre la que se construye ZigBee consiste en disminuir en gran medida los tiempos de respuesta (la velocidad de transmisión puede ser modificada de acuerdo a la aplicación que se requiera) lo cual permite operaciones con un menor consumo energético y un tiempo de respuesta casi instantáneo con un retardo de milisegundos lo cual para el sistema y monitoreo de los sensores es muy adecuado.

La tecnología ZigBee es ideal para aplicaciones como monitoreo y control para el uso industrial de plantas de proceso, así como para la automatización del hogar, control de energía y medioambiente.

Las ventajas de este sistema instrumentado mediante módulos con tecnología Zigbee es que podemos monitorear el sistema desde múltiples sistemas o computadoras ya que podemos tener conectados hasta 65.000 dispositivos a la vez, son baratos, se puede proveer conexiones seguras entre dispositivos por su capacidad de empaquetar datos con una

estructura definida y su implementación a nivel hardware llega a ser más sencilla. Para poder implementar los puntos anteriores con otro tipo de tecnología como Bluetooth, Wi-Fi o cualquier otro dispositivo de radiofrecuencia podría llegar a ser muy costoso y difícil de implementar.

Cabe mencionar que este sistema de nivel de líquido es un modelo didáctico que por las dimensiones de su tamaño se puede denominar de escala industrial y el fin de su construcción es para practicar diferentes técnicas de control e instrumentación dentro de la facultad, por lo que sus ventajas en cuanto al diseño del sistema con respecto a algún sistema del tipo industrial no se requieren definir. De esta manera el sistema diseñado representa una herramienta didáctica importante para el área de control analógico e instrumentación.

La tecnología de ZigBee no acabará con otras tecnologías inalámbricas ya consolidadas, como Wi-Fi y Bluetooth, sino que convivirá con ellas y encontrará sus propios nichos de mercado.

VIII. AGRADECIMIENTOS

Los autores desean agradecer a la Coordinación de la Investigación Científica de la Universidad Michoacana de San Nicolás de Hidalgo el apoyo económico recibido a través del proyecto de investigación "Diseño de algoritmos de control para sistemas de nivel de líquidos", de donde se desprende el presente artículo.

IX. REFERENCIAS

- [1] J. Gutiérrez, M.A. Porta-Gándara, "Medidor ultrasónico de nivel de agua para estanques", Ingeniería Investigación y Tecnología, Vol. VII-4, 2006
- [2] B. Cadena V.M., "Construcción de un módulo para el control de nivel de líquidos", XIX Jornadas en Ingeniería Eléctrica y Electrónica, Vol. 19, 2005.
- [3] Página del fabricante (Digi, antes MaxStream) - <http://www.digi.com/products/wireless/point-multipoint/xbee-series1-module.jsp>
- [4] Procedimiento para actualizar el firmware - http://itp.nyu.edu/~raf275/meshnetworking/XBee/XBee_firmware_upgrade.html
- [5] Libro de Tom Igoe *Making Things Talk* - <http://itp.nyu.edu/~raf275/cgi-bin/mt/search.cgi?IncludeBlogs=3&search=xbee>
- [6] José A. Chaves, Hugo B. Cano, Esteban Giraldo, "Propuesta de Aplicación "Medición del consumo de Agua Domiciliaria" Utilizando Tecnología Inalámbrica Zigbee", Scientia et Technica Año XIV, No 39, Septiembre de 2008. Universidad Tecnológica de Pereira. ISSN 0122-1701
- [7] Adams, Jon; Bob Heile (2005-10). Busy as a ZigBee. [IEEE]
- [8] Características del producto:



http://ftp1.digi.com/support/documentation/90000991_B.pdf

[9] Manual del producto:

http://ftp1.digi.com/support/documentation/90000982_B.pdf

[10] El manual del usuario de MaxStream

http://www.makingthings.com/resources/datasheets/manual_xb_oem-rfmodules_802.15.4.pdf

X. BIOGRAFÍAS

Salvador Ramírez Zavala nació en Morelia, Michoacán. Recibió el grado de Ingeniero Electricista en la Universidad Michoacana de San Nicolás de Hidalgo, el grado de Maestro en Ingeniería Eléctrica en la misma Institución en 1990 y 1998 respectivamente. Sus áreas de interés son Electrónica de Potencia, Robótica, Control e Instrumentación.

Isidro Ignacio Lázaro Castillo nació en Córdoba, Veracruz, México. Recibió el grado de Ingeniero Electricista en la Universidad Michoacana de San Nicolás de Hidalgo, el grado de maestro en Ingeniería Eléctrica en la misma Institución en 1992 y 1999 respectivamente. Actualmente es Profesor Investigador de tiempo completo de la misma Facultad. Ha publicado varios artículos científicos y técnicos en congresos nacionales, internacionales y revistas. Autor del libro Ingeniería de Sistemas de Control Continuo. Sus áreas de interés son Calidad de la Energía, Electrónica de Potencia, Control e Instrumentación.

Alejandro Barajas Castillo nació en Morelia, Michoacán, el 24 de octubre de 1988. Es estudiante del noveno semestre de la carrera de Ingeniería en Electrónica en la Facultad de Ingeniería Eléctrica de la Universidad Michoacana de San Nicolás de Hidalgo. Sus intereses son en el ramo de la Electrónica Digital, Comunicaciones Inalámbricas, Robótica, Micro-controladores y Programación.

Apéndice C

CARACTERES ASCII



The ASCII Character Set

The American Standard Code for Information Interchange or ASCII assigns values between 0 and 255 for upper and lower case letters, numeric digits, punctuation marks and other symbols. ASCII characters can be split into the following sections:

- ❑ 0 – 31 Control codes
- ❑ 32 – 127 Standard, implementation-independent characters
- ❑ 128 – 255 Special symbols, international character sets – generally, non-standard characters.

Control Codes : ASCII Characters 0 - 31

The following table lists and describes the first 32 ASCII characters, often referred to as control codes. The columns show the decimal and hexadecimal ASCII values for each code along with their abbreviated and full names. Descriptions are given to those most in use today.

Decimal	Octal	Hexadecimal	Code	Description
000	000	00	NUL	Null
001	001	01	SOH	Start Of Heading
002	002	02	STX	Start of TeXt
003	003	03	ETX	End of TeXt
004	004	04	EOT	End Of Transmission
005	005	05	ENQ	ENQuiry
006	006	06	ACK	ACKnowledge

Table continued on following page

Decimal	Octal	Hexadecimal	Code	Description
007	007	07	BEL	BELl. Caused teletype machines to ring a bell. Causes a beep in many common terminals and terminal emulation programs.
008	010	08	BS	BackSpace. Moves the cursor move backwards (left) one space.
009	011	09	HT	Horizontal Tab. Moves the cursor right to the next tab stop. The spacing of tab stops is dependent on the output device, but is often either 8 or 10 characters wide.
010	012	0A	LF	Line Feed. Moves the cursor to a new line. On Unix systems, moves to a new line AND all the way to the left.
011	013	0B	VT	Vertical Tab
012	014	0C	FF	Form Feed. Advances paper to the top of the next page (if the output device is a printer).
013	015	0D	CR	Carriage Return. Moves the cursor all the way to the left, but does not advance to the next line.
014	016	0E	SO	Shift Out
015	017	0F	SI	Shift In
016	020	10	DLE	Data Link Escape
017	021	11	DC1	Device Control 1
018	022	12	DC2	Device Control 2
019	023	13	DC3	Device Control 3
020	024	14	DC4	Device Control 4
021	025	15	NAK	Negative AcKnowledge
022	026	16	SYN	SYNchronous idle
023	027	17	ETB	End of Transmission Block
024	030	18	CAN	CANcel
025	031	19	EM	End of Medium
026	032	1A	SUB	SUBstitute
027	033	1B	ESC	ESCape
028	034	1C	FS	File Separator

Decimal	Octal	Hexadecimal	Code	Description
029	035	1D	GS	Group Separator
030	036	1E	RS	Record Separator
031	037	1F	US	Unit Separator

The Standard ASCII Characters : 32 - 127

ASCII Characters 32 - 127 are the standard, implementation-independent alphanumeric characters we work with every day. The tables below show the characters along with both their decimal and hexadecimal ASCII values.

Characters 32 - 64

The first table, which contains characters 32 - 64, contains the majority of the standard symbolic characters and the numbers from zero to nine.

Decimal	Octal	Hexadecimal	Character	Decimal	Octal	Hexadecimal	Character
032	040	20	Space	049	061	31	1
033	041	21	!	050	062	32	2
034	042	22	"	051	063	33	3
035	043	23	#	052	064	34	4
036	044	24	\$	053	065	35	5
037	045	25	%	054	066	36	6
038	046	26	&	055	067	37	7
039	047	27	'	056	070	38	8
040	050	28	(	057	071	39	9
041	051	29	)	058	072	3A	:
042	052	2A	*	059	073	3B	;
043	053	2B	+	060	074	3C	<
044	054	2C	,	061	075	3D	=
045	055	2D	-	062	076	3E	>
046	056	2E	.	063	077	3F	?
047	057	2F	/	064	100	40	@
048	060	30	0				

Characters 65 - 127

The second table, which contains characters 65 - 127, contains the standard Latin alphabet characters both lower and upper case, separated only by a few characters at 91 - 96 and 123 - 127.

Decimal	Octal	Hexadecimal	Character	Decimal	Octal	Hexadecimal	Character
065	101	41	A	097	141	61	a
066	102	42	B	098	142	62	b
067	103	43	C	099	143	63	c
068	104	44	D	100	144	64	d
069	105	45	E	101	145	65	e
070	106	46	F	102	146	66	f
071	107	47	G	103	147	67	g
072	110	48	H	104	150	68	h
073	111	49	I	105	151	69	i
074	112	4A	J	106	152	6A	j
075	113	4B	K	107	153	6B	k
076	114	4C	L	108	154	6C	l
077	115	4D	M	109	155	6D	m
078	116	4E	N	110	156	6E	n
079	117	4F	O	111	157	6F	o
080	120	50	P	112	160	70	p
081	121	51	Q	113	161	71	q
082	122	52	R	114	162	72	r
083	123	53	S	115	163	73	s
084	124	54	T	116	164	74	t
085	125	55	U	117	165	75	u
086	126	56	V	118	166	76	v
087	127	57	W	119	167	77	w
088	130	58	X	120	170	78	x
089	131	59	Y	121	171	79	y
090	132	5A	Z	122	172	7A	z

Decimal	Octal	Hexadecimal	Character	Decimal	Octal	Hexadecimal	Character
091	133	5B	[	123	173	7B	{
092	134	5C	\	124	174	7C	
093	135	5D	]	125	175	7D	}
094	136	5E	^	126	176	7E	~
095	137	5F	_	127	177	7F	delete
096	140	60	'				

The Non-Standard ASCII Characters : 128 - 255

The second half of the ASCII table holds the non-standard extension set of characters which may vary depending which computer system you may be using. One common – but in no way definitive – example of this extended set is as follows.

Characters 128 - 191

This first table contains characters 128 - 191, abstract symbols that appear in text from time to time.

Decimal	Octal	Hexadecimal	Character	Decimal	Octal	Hexadecimal	Character
128	200	80	•	160	240	A0	non-breaking space
129	201	81	•	161	241	A1	ı
130	202	82	,	162	242	A2	¢
131	203	83	<i>f</i>	163	243	A3	£
132	204	84	„	164	244	A4	□
133	205	85	...	165	245	A5	¥
134	206	86	†	166	246	A6	‡
135	207	87	‡	167	247	A7	\$
136	210	88	^	168	250	A8	..
137	211	89	‰	169	251	A9	©
138	212	8A	Š	170	252	AA	ª
139	213	8B	<	171	253	AB	«

Table continued on following page

Decimal	Octal	Hexadecimal	Character	Decimal	Octal	Hexadecimal	Character
140	214	8C	Œ	172	254	AC	¬
141	215	8D	•	173	255	AD	-
142	216	8E	Ž	174	256	AE	®
143	217	8F	•	175	257	AF	-
144	220	90	•	176	260	B0	°
145	221	91	'	177	261	B1	±
146	222	92	'	178	262	B2	²
147	223	93	"	179	263	B3	³
148	224	94	•	180	264	B4	˘
149	225	95	•	181	265	B5	µ
150	226	96	—	182	266	B6	¶
151	227	97	—	183	267	B27	·
152	230	98	˘	184	270	B8	¸
153	231	99	™	185	271	B9	¹
154	232	9A	§	186	272	BA	º
155	233	9B	›	187	273	BB	»
156	234	9C	œ	188	274	BC	¼
157	235	9D	•	189	275	BD	½
158	236	9E	ÿ	190	276	BE	¾
159	237	9F	ÿ	191	277	BF	¿

Characters 192 - 255

The second table contains characters 192 - 255, variously accented alphabetical characters.

Decimal	Octal	Hexadecimal	Character	Decimal	Octal	Hexadecimal	Character
192	300	C0	À	224	340	E0	à
193	301	C1	Á	225	341	E1	á
194	302	C2	Â	226	342	E2	â
195	303	C3	Ã	227	343	E3	ã
196	304	C4	Ä	228	344	E4	ä

Decimal	Octal	Hexadecimal	Character	Decimal	Octal	Hexadecimal	Character
197	305	C5	Å	229	345	E5	å
198	306	C6	Æ	230	346	E6	æ
199	307	C7	Ç	231	347	E7	ç
200	310	C8	È	232	350	E8	è
201	311	C9	É	233	351	E9	é
202	312	CA	Ê	234	352	EA	ê
203	313	CB	Ë	235	353	EB	ë
204	314	CC	Ì	236	354	EC	ì
205	315	CD	Í	237	355	ED	í
206	316	CE	Î	238	356	EE	î
207	317	CF	Ï	239	357	EF	ï
208	320	D0	Ð	240	360	F0	ð
209	321	D1	Ñ	241	361	F1	ñ
210	322	D2	Ò	242	362	F2	ò
211	323	D3	Ó	243	363	F3	ó
212	324	D4	Ô	244	364	F4	ô
213	325	D5	Ö	245	365	F5	ö
214	326	D6	Ö	246	366	F6	ö
215	327	D7	×	247	367	F7	÷
216	330	D8	Ø	248	370	F8	ø
217	331	D9	Ù	249	371	F9	ù
218	332	DA	Ú	250	372	FA	ú
219	333	DB	Û	251	373	FB	û
220	334	DC	Ü	252	374	FC	ü
221	335	DD	Ý	253	375	FD	ý
222	336	DE	Þ	254	376	FE	þ
223	337	DF	ß	255	377	FF	

Apéndice D

CARACTERÍSTICAS DE LA ELECTROVÁLVULA, SENSOR DE ULTRASONIDO,
MICROCONTROLADORES, MÓDULO GSM, MÓDULO XBEE® Y ESQUEMÁTICOS DE LAS
TARJETAS REALIZADAS

Apéndice D.1

CARACTERÍSTICAS DE LA ELECTROVÁLVULA HONEYWELL ML6984

ML6984 Direct Coupled Valve Actuator

PRODUCT DATA



APPLICATION

The ML6984 is a self-contained, self-adjusting, linear motorized linkage that mounts directly onto V5011 two-way or V5013 three-way valves and provides up to 19 mm (3/4") of linear travel (stem lift). For use with low voltage 3-wire SPDT Series 20 (on-off), Series 60 (floating) electro-mechanical (dry contacts) or electronic (triac) output controllers.

FEATURES

- Allows the use of one common transformer power supply for multiple actuators and controllers
- Self-contained, motorized valve linkage
- Linkage self-adjusts to valve stroke of up to 19mm (3/4")
- Multipose mounting
- Strong valve seat closing force 710 Newton (160 lbs.)
- Compact size for easy installation in confined area
- One device for either Vac or Vdc power supply application
- Electronic current sensing provides internal protection and positive full closing force
- Field-addable auxiliary switches available
- Compatible with 3-wire or 5-wire control systems

Contents

Specifications.....	2
Order information.....	2
Installation.....	3
Wiring Schematics.....	5
Operation and Checkout.....	6





-  Allow 0.5 amps maximum for each device. Actuators and controller can share same transformer providing the VA rating of the transformer is not exceeded and proper phasing is observed.
-  SPDT controller can be a low voltage Series 20 "on-off " or Series 60 "floating" (center-off) type.
-  For multiple actuators controlled by a common controller in parallel, these actuators must be wired so that they all travel in the same direction.

5

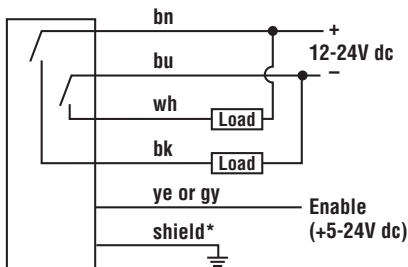
Apéndice D.2

CARACTERÍSTICAS DEL SENSOR DE ULTRASONIDO BANNER Q45U

U-GAGE™ Q45U Ultrasonic Sensors with Bipolar Discrete Outputs

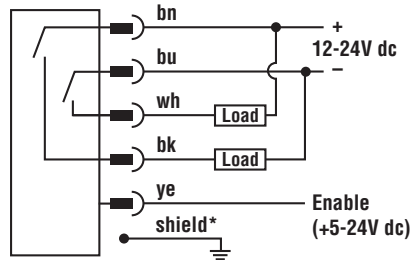
Q45U Series Hookup Diagrams

Q45U Sensor with Attached Cable

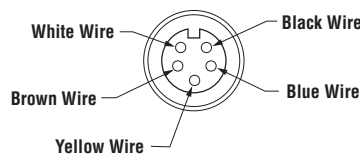


*It is recommended that the shield wire be connected to earth ground or dc common.

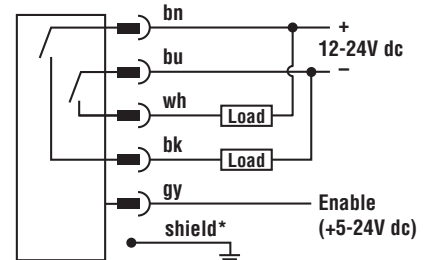
**Q45U Sensor with 5-Pin Mini-Style QD
("Q" model Suffix)**



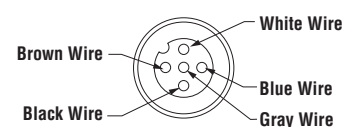
**5-Pin Mini-Style Pin-out
(Cable Connector Shown)**



**Q45U Sensor with 5-Pin Euro-Style QD
("Q6" model Suffix)**

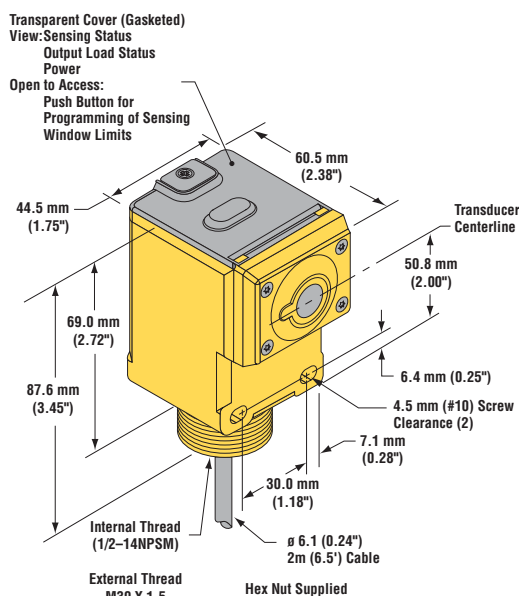


**5-Pin Euro-Style Pin-out
(Cable Connector Shown)**

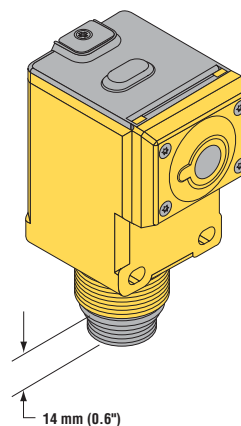


Q45U Series Dimensions

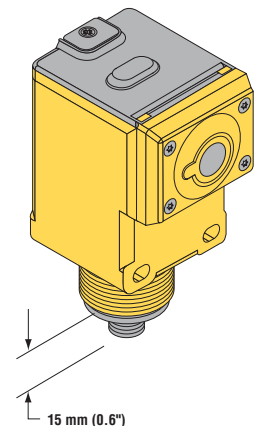
Q45U Sensor with Attached Cable



**Q45U Sensor with 5-Pin Mini-Style QD
("Q" model Suffix)**



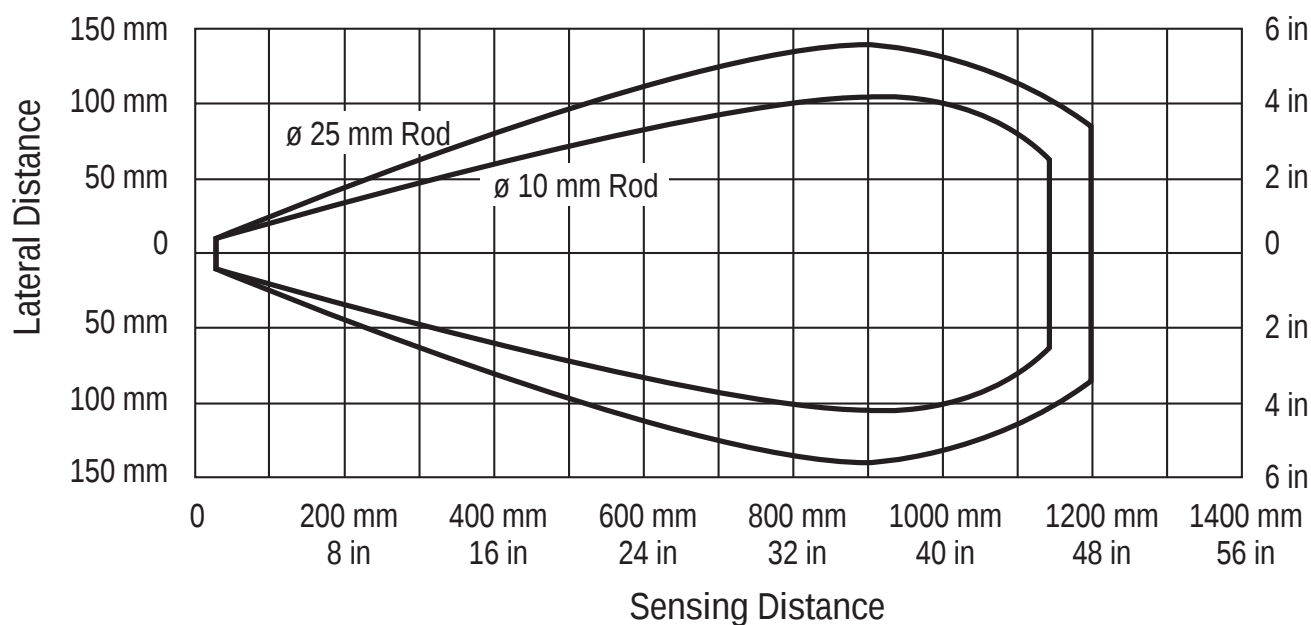
**Q45U Sensor with 5-Pin Euro-Style QD
("Q6" model Suffix)**



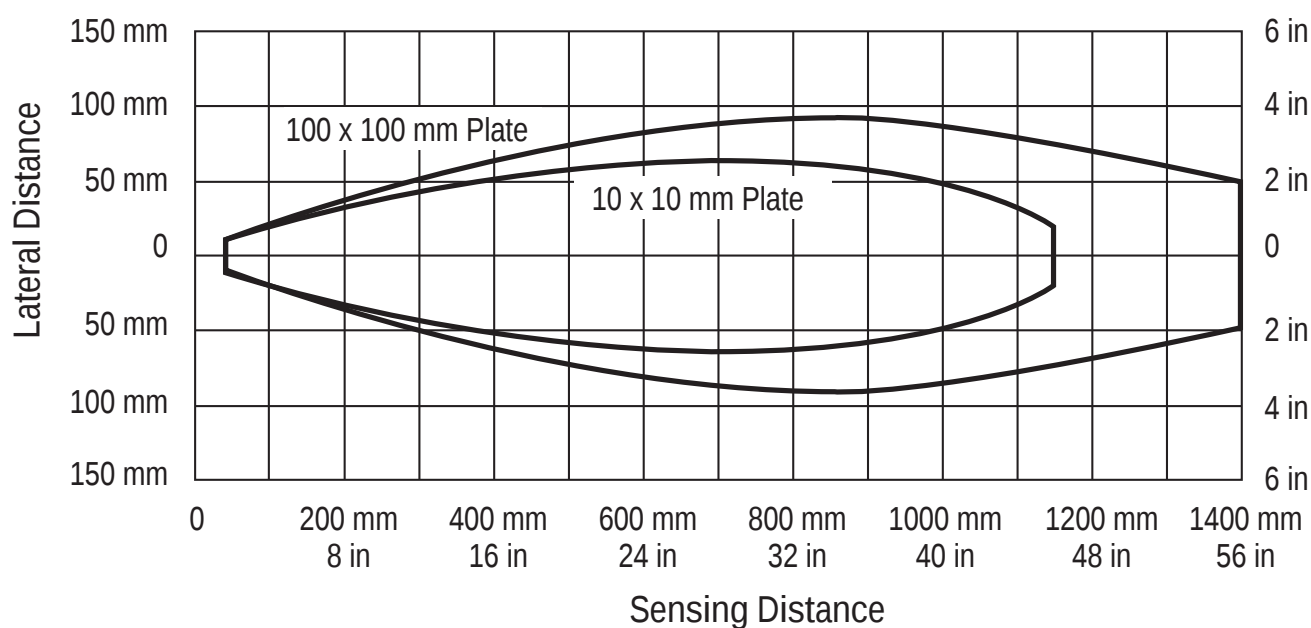
U-GAGE™ Q45U Ultrasonic Sensors with Bipolar Discrete Outputs

Q45U Response Curves

Q45U Effective Beam with Rod Target (Typical)



Q45U Effective Beam with Plate Target (Typical)



Apéndice D.3

CARACTERÍSTICAS DEL PIC18F4550 Y PIC18F2550

**MICROCHIP****PIC18F2455/2550/4455/4550**

28/40/44-Pin, High-Performance, Enhanced Flash, USB Microcontrollers with nanoWatt Technology

Universal Serial Bus Features:

- USB V2.0 Compliant
- Low Speed (1.5 Mb/s) and Full Speed (12 Mb/s)
- Supports Control, Interrupt, Isochronous and Bulk Transfers
- Supports up to 32 Endpoints (16 bidirectional)
- 1-Kbyte Dual Access RAM for USB
- On-Chip USB Transceiver with On-Chip Voltage Regulator
- Interface for Off-Chip USB Transceiver
- Streaming Parallel Port (SPP) for USB streaming transfers (40/44-pin devices only)

Power-Managed Modes:

- Run: CPU on, peripherals on
- Idle: CPU off, peripherals on
- Sleep: CPU off, peripherals off
- Idle mode currents down to 5.8 μ A typical
- Sleep mode currents down to 0.1 μ A typical
- Timer1 Oscillator: 1.1 μ A typical, 32 kHz, 2V
- Watchdog Timer: 2.1 μ A typical
- Two-Speed Oscillator Start-up

Flexible Oscillator Structure:

- Four Crystal modes, including High Precision PLL for USB
- Two External Clock modes, up to 48 MHz
- Internal Oscillator Block:
 - 8 user-selectable frequencies, from 31 kHz to 8 MHz
 - User-tunable to compensate for frequency drift
- Secondary Oscillator using Timer1 @ 32 kHz
- Dual Oscillator options allow microcontroller and USB module to run at different clock speeds
- Fail-Safe Clock Monitor:
 - Allows for safe shutdown if any clock stops

Peripheral Highlights:

- High-Current Sink/Source: 25 mA/25 mA
- Three External Interrupts
- Four Timer modules (Timer0 to Timer3)
- Up to 2 Capture/Compare/PWM (CCP) modules:
 - Capture is 16-bit, max. resolution 5.2 ns ($T_{CY}/16$)
 - Compare is 16-bit, max. resolution 83.3 ns (T_{CY})
 - PWM output: PWM resolution is 1 to 10-bit
- Enhanced Capture/Compare/PWM (ECCP) module:
 - Multiple output modes
 - Selectable polarity
 - Programmable dead time
 - Auto-shutdown and auto-restart
- Enhanced USART module:
 - LIN bus support
- Master Synchronous Serial Port (MSSP) module supporting 3-wire SPI (all 4 modes) and I²C[™] Master and Slave modes
- 10-bit, up to 13-channel Analog-to-Digital Converter module (A/D) with Programmable Acquisition Time
- Dual Analog Comparators with Input Multiplexing

Special Microcontroller Features:

- C Compiler Optimized Architecture with optional Extended Instruction Set
- 100,000 Erase/Write Cycle Enhanced Flash Program Memory typical
- 1,000,000 Erase/Write Cycle Data EEPROM Memory typical
- Flash/Data EEPROM Retention: > 40 years
- Self-Programmable under Software Control
- Priority Levels for Interrupts
- 8 x 8 Single-Cycle Hardware Multiplier
- Extended Watchdog Timer (WDT):
 - Programmable period from 41 ms to 131s
- Programmable Code Protection
- Single-Supply 5V In-Circuit Serial Programming[™] (ICSP[™]) via two pins
- In-Circuit Debug (ICD) via two pins
- Optional dedicated ICD/ICSP port (44-pin devices only)
- Wide Operating Voltage Range (2.0V to 5.5V)

Device	Program Memory		Data Memory		I/O	10-Bit A/D (ch)	CCP/ECCP (PWM)	SPP	MSSP		EAUSART	Comparators	Timers 8/16-Bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)					SPI	Master I ² C [™]			
PIC18F2455	24K	12288	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F2550	32K	16384	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F4455	24K	12288	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3
PIC18F4550	32K	16384	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3

Apéndice D.4

CARACTERÍSTICAS DEL MÓDULO GSM SM5100B

Power consumption	Off mode: <100uA Sleep mode: <2.0mA Idle mode: <7.0mA (average) Communication mode: 350 mA (average,GSM) Communication mode: 2000mA (Typical peak during TX slot,GSM)
Li-ion Battery charging management and interface (OPTION)	Li-ion Battery charging management is included. The charger interface is provided on 60-pin connector. (only for 3.7V Li-ion Battery)
Frequency bands	EGSM900 +GSM850+ DCS1800+PCS1900
Transmit power	Class 4 (2W) for EGSM900/GSM850 Class 1 (1W) for DCS1800/PCS1900
Supported SIM card	3V/1.8V SIM card. (auto recognise)
Keyboard interface	4x6 keyboard interface is provided
UART0 interface with flow control	Up to 460 kbps Full hardware flow control signals (+3.0V) are provided on 60 pins.
UART1 interface without flow control	2-Wire UART interface Up to 460 kbps
LCD interface	Support standard SPI interface

1.4 FUNCTIONAL DESCRIPTION

1.4.1 RF Functionalities

The RF part of this module converts RF signals to baseband for receiver chain and translates base band signals into RF frequency spectrum.

The operating frequencies are:

Rx (EGSM 850): 869 to 894MHz

Tx (EGSM 850): 824 to 849MHz

Rx (EGSM 900): 925 to 960MHz

Tx (EGSM 900): 880 to 915MHz

Rx (DCS 1800): 1805 to 1880MHz

Tx (DCS 1800): 1710 to 1785MHz

Rx (PCS 1900): 1930 to 1990MHz

Tx (PCS 1900): 1850 to 1910MHz

1.4.2 Baseband Functionalities

The baseband part of SM5210 is composed of a SPREADTRUM's SC6600D chip. This chipset is using 0.18μm mixed signal CMOS technology which allows massive integration as well as low power consumption. SC6600D provides single-chip solution to wireless Quad-band telephone handsets and data modems confirming to the EGSM 900, GSM 850, DCS 1800 and PCS 1900.

1.4.3 Memory Functionalities

The memory used in this module is a combination of 32 Megabit (2M x 16-Bit) CMOS 3.0V Volt-only, simultaneous operation flash memory and 4 Megabit (256K x 16-Bit) static RAM.

2 INTERFACE AND FUNCTIONALITIES

2.1 POWER SUPPLY

2.1.1 Power Supply Description

The power supply is one of the key issues in the designing GSM terminals. Due to the 577us radio burst emission in GSM every 4.615ms, power supply must be able to deliver high current peaks in a short time. During these peaks, ripples and drops on the supply voltage must not exceed a certain limit.

Table 1: VBAT Pin Description

	Pin number	V _{MIN}	V _{NOM}	V _{MAX}	Ripple max
VBAT (V)	1,3,5,7	3.3V (*)	3.6V	4.2V (**)	50mV _{PP} for freq<200KHz 2mV _{PP} for freq>200KHz

(*): This value has to be guaranteed during the burst (with 2.0A Peak in GSM or GPRS mode).

(**): max operation Voltage Stationary Wave Ratio.

When supplying the module with a battery, the total impedance (battery + protections + PCBS) should be less than 150mOhms.

Table 2: GND Pin Description

	Pin number	Description
GND	9,11,13,15,17 Shielding can install pin(*)	Ground

2.1.2 Power Consumption

Table 3: Power consumption in OFF mode

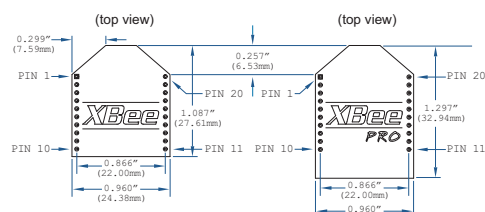
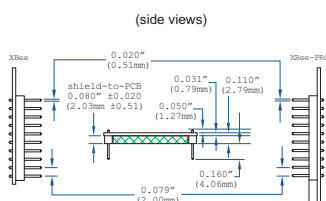
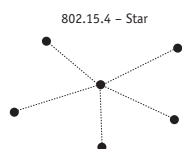
Mode	Condition	I(Typical)	I(Max)
Off mode	off	50 uA	100uA

Apéndice D.5

CARACTERÍSTICAS DE LOS MÓDULOS XBEE® MULTIPPOINT RF MODULES

Platform	XBee® 802.15.4 (Series 1)	XBee-PRO® 802.15.4 (Series 1)	XBee-PRO® XSC
Performance			
RF Data Rate	250 kbps	250 kbps	10 kbps / 9.6 kbps
Indor/Urban Range	100 ft (30 m)	300 ft (100 m)	Up to 1200 ft (370 m)
Outdoor/RF Line-of-Sight Range	300 ft (100 m)	1 mi (1.6 km)	Up to 6 mi (9.6 km)
Transmit Power	1 mW (+0 dBm)	60 mW (+18 dBm)*	100 mW (+20 dBm)
Receiver Sensitivity (1% PER)	-92 dBm	-100 dBm	-106 dBm
Features			
Serial Data Interface	3.3V CMOS UART	3.3V CMOS UART	3.3V CMOS UART (5V Tolerant)
Configuration Method	API or AT Commands, local or over-the-air	API or AT Commands, local or over-the-air	AT Commands
Frequency Band	2.4 GHz	2.4 GHz	902 MHz to 928 MHz
Interference Immunity	DSSS (Direct Sequence Spread Spectrum)	DSSS (Direct Sequence Spread Spectrum)	FHSS (Frequency Hopping Spread Spectrum)
Serial Data Rate	1200 bps - 250 kbps	1200 bps - 250 kbps	1200 bps - 57.6 kbps
ADC Inputs	(6) 10-bit ADC inputs	(6) 10-bit ADC inputs	None
Digital I/O	8	8	None
Antenna Options	Chip, Wire Whip, U.FL, & RPSMA	Chip, Wire Whip, U.FL, & RPSMA	Wire Whip, U.FL, RPSMA
Networking & Security			
Encryption	128-bit AES	128-bit AES	No
Reliable Packet Delivery	Retries/Acknowledgments	Retries/Acknowledgments	Retries/Acknowledgements
IDs and Channels	PAN ID, 64-bit IEEE MAC, 16 Channels	PAN ID, 64-bit IEEE MAC, 12 Channels	PAN ID, 32-bit Address, 7 Channels
Power Requirements			
Supply Voltage	2.8 - 3.4VDC	2.8 - 3.4VDC	3.0 - 3.6VDC
Transmit Current	45 mA @ 3.3VDC	215 mA @ 3.3VDC	265 mA typical
Receive Current	50 mA @ 3.3VDC	55 mA @ 3.3VDC	65 mA typical
Power-Down Current	<10 uA @ 25° C	<10 uA @ 25° C	45 uA pin Sleep
Regulatory Approvals			
FCC (USA)	OUR-XBEE	OUR-XBEEPRO	MCQ-XBEEEXSC
IC (Canada)	4214A-XBEE	4214A-XBEEPRO	1846A-XBEEEXSC
ETSI (Europe)	Yes	Yes* Max TX 10 mW	No
C-TICK Australia	Yes	Yes	No
Telec (Japan)	Yes	Yes*	No

* XBee-PRO 802.15.4 TX Power restricted to 10 mW in Europe and Japan.



Please visit www.digi.com for part numbers.

DIGI SERVICE AND SUPPORT - You can purchase with confidence knowing that Digi is here to support you with expert technical support and a one-year warranty. www.digi.com/support

**WHEN
RELIABILITY
MATTERS™**

Digi International
11001 Bren Road E.
Minnetonka, MN 55343
U.S.A.
PH: 877-912-3444
952-912-3444
FX: 952-912-4952
email: info@digi.com

**Digi International
France**
31 rue des Poissonniers
92200 Neuilly sur Seine
PH: +33-1-55-61-98-98
FX: +33-1-55-61-98-99
www.digi.fr

**Digi International
KK**
NES Building South 8F
22-14 Sakuragaoka-cho,
Shibuya-ku
Tokyo 150-0031, Japan
PH: +81-3-5428-0261
FX: +81-3-5428-0262
www.digi-intl.co.jp

**Digi International
(HK) Limited**
Suite 1703-05, 17/F,
K Wah Centre
191 Java Road
North Point, Hong Kong
PH: +852-2833-1008
FX: +852-2572-9989
www.digi.cn

Digi International, the leader in device networking for business, develops reliable products and technologies to connect and securely manage local or remote electronic devices over the network or via the web. With over 20 million ports shipped worldwide since 1985, Digi offers the highest levels of performance, flexibility and quality.

www.digi.com

© 2006-2008 Digi International Inc.

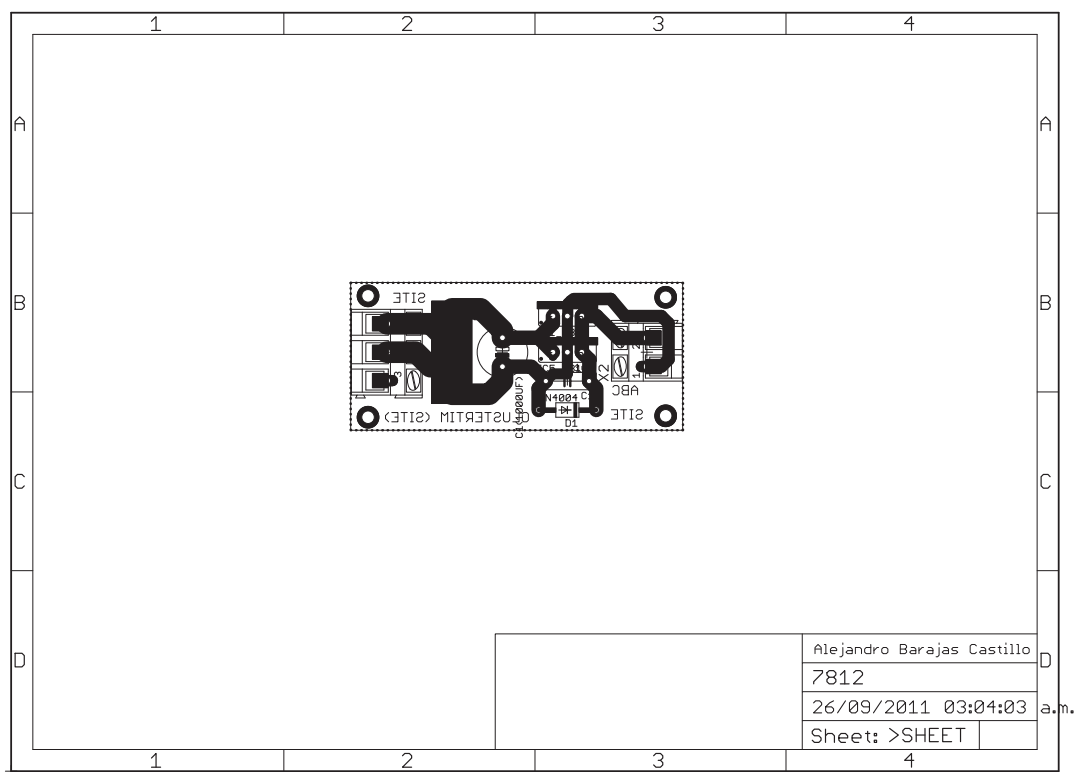
All rights reserved. Digi, Digi International, the Digi logo, the When Reliability Matters logo, XBee and XBee-PRO are trademarks or registered trademarks of Digi International Inc. in the United States and other countries worldwide. All other trademarks are the property of their respective owners.

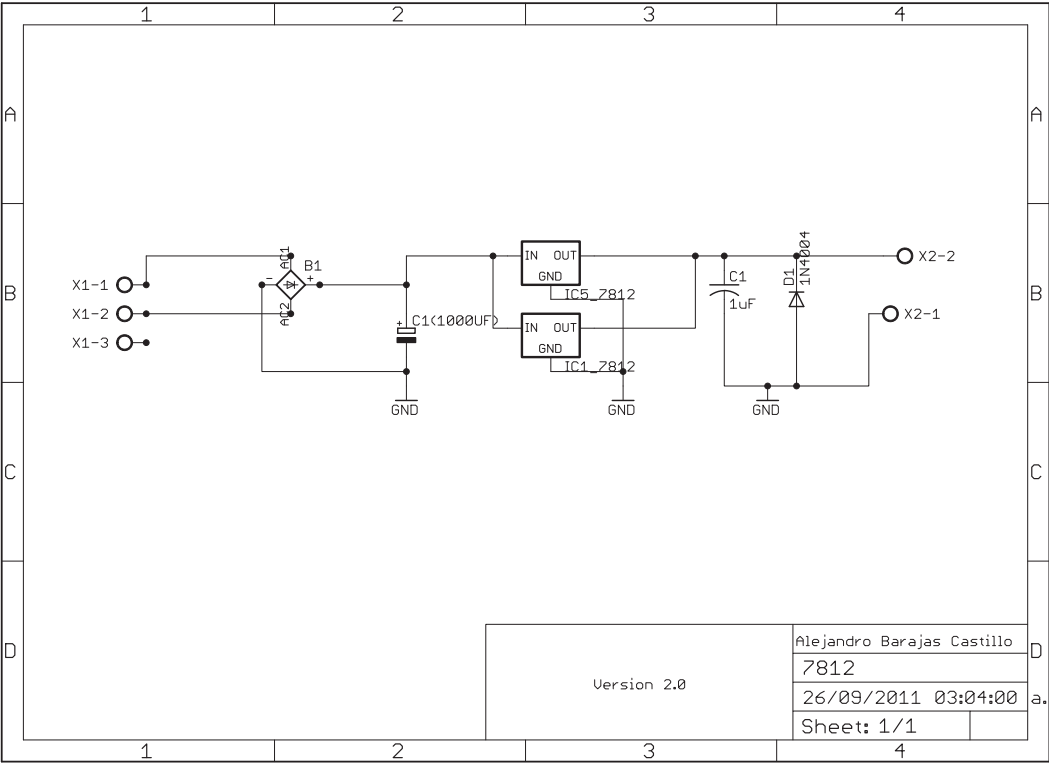
91001412
B1/308



Apéndice D.6

ESQUEMÁTICOS DE LA FUENTE DE 12 VOLTS DE LOS SENSORES DE ULTRASONIDO





Version 2.0

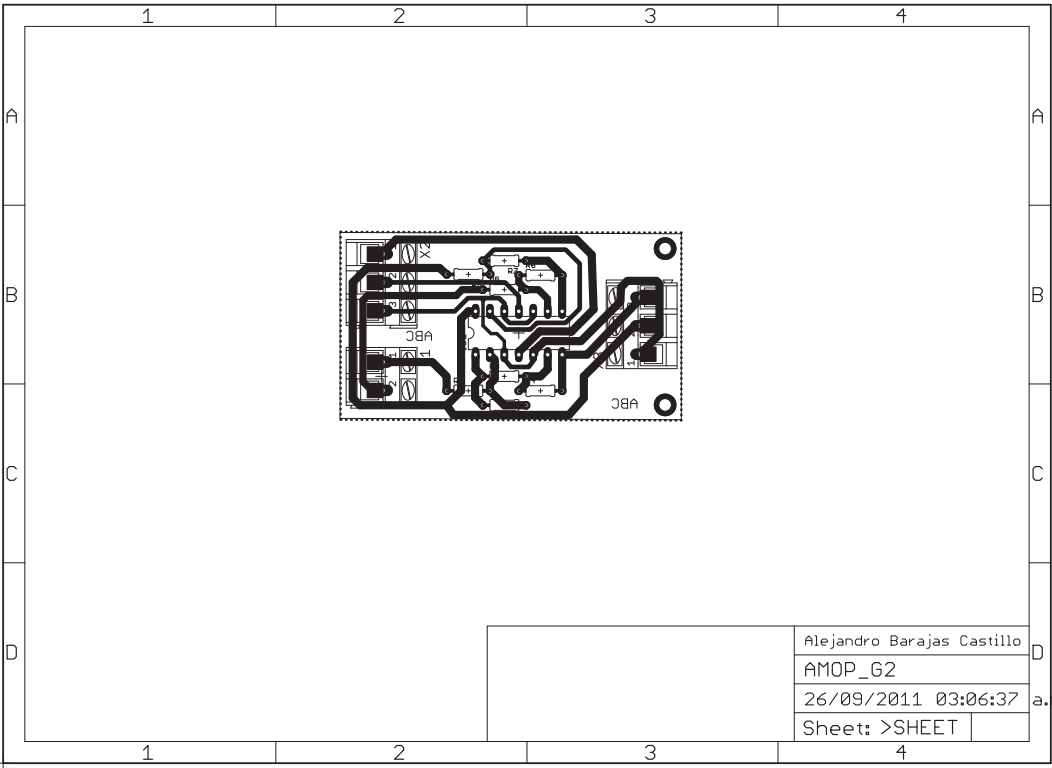
Alejandro Barajas Castillo	
7812	
26/09/2011 03:04:00	
Sheet: 1/1	

a.m.

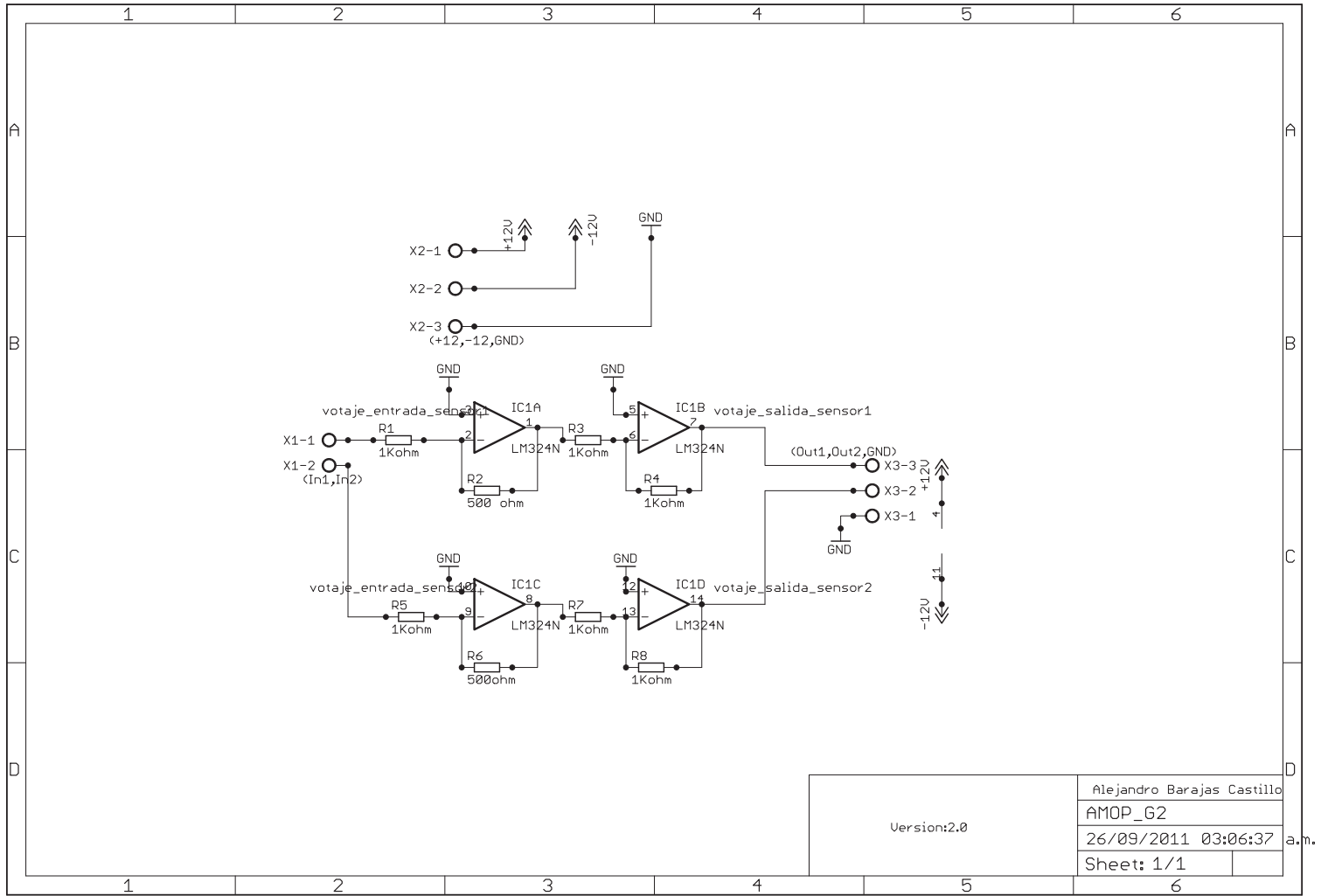
Apéndice D.7

ESQUEMÁTICOS DEL ARREGLO DE AMPLIFICADORES OPERACIONALES PARA REDUCIR

EL VOLTAGE DE LOS SENSORES DE ULTRASONIDO A UN RANGO DE 0 A 5 VOLTS.

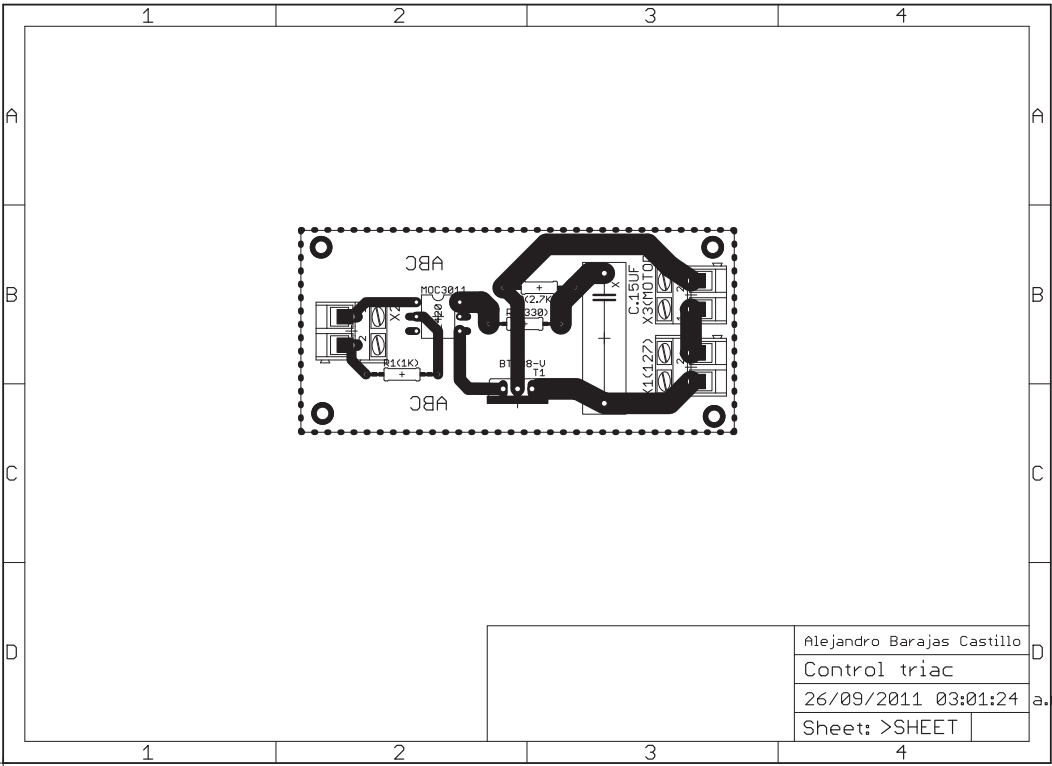


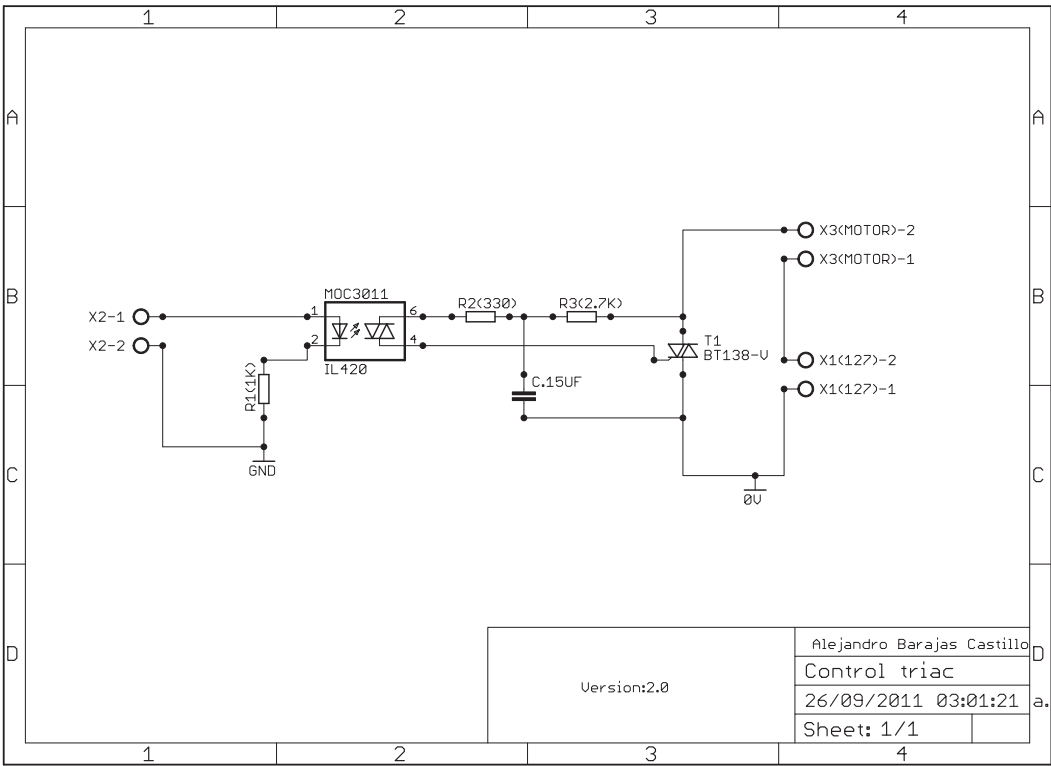
a.m.



Apéndice D.8

ESQUEMÁTICOS DEL CONTROLADOR DEL MOTOR

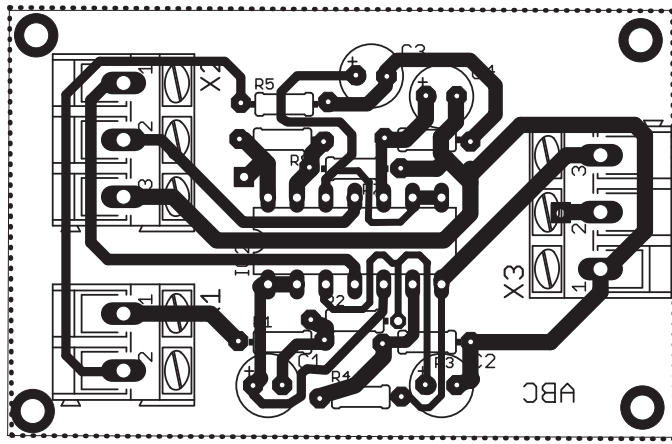


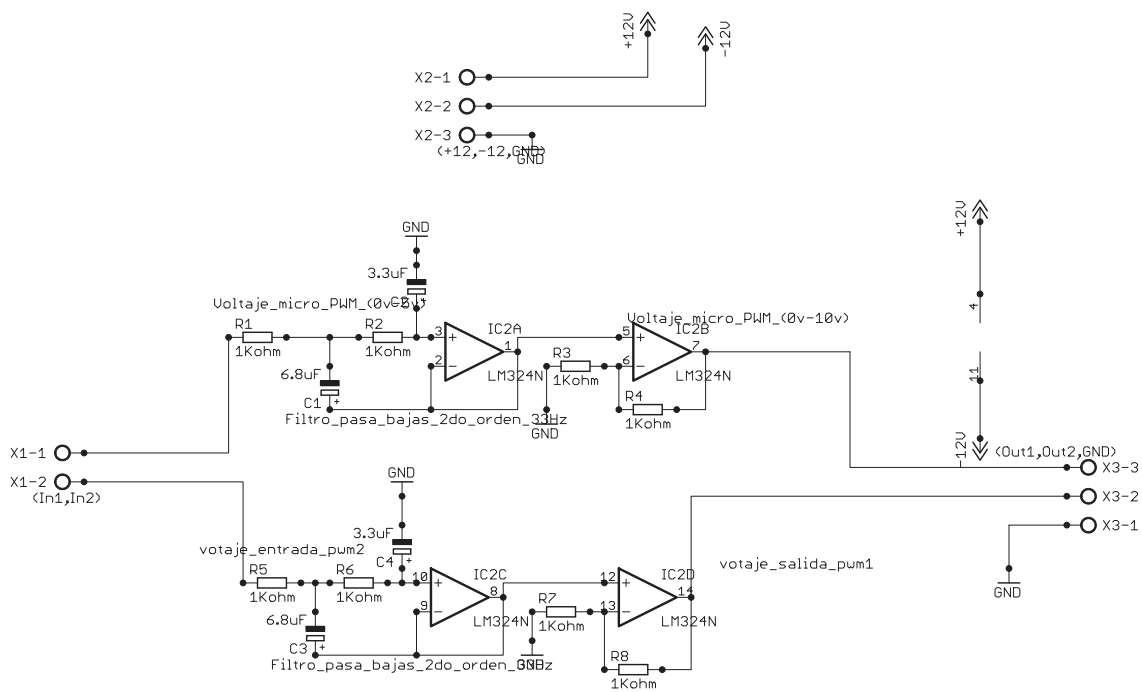


Apéndice D.9

ESQUEMÁTICOS DEL FILTRO BUTTERWORTH PASA BAJAS DE SEGUNDO ORDEN CON

UNA FRECUENCIA DE CORTE DE 33Hz





Alejandro Barajas Castillo

TITLE: filtro_tanques

Document Number: 1212

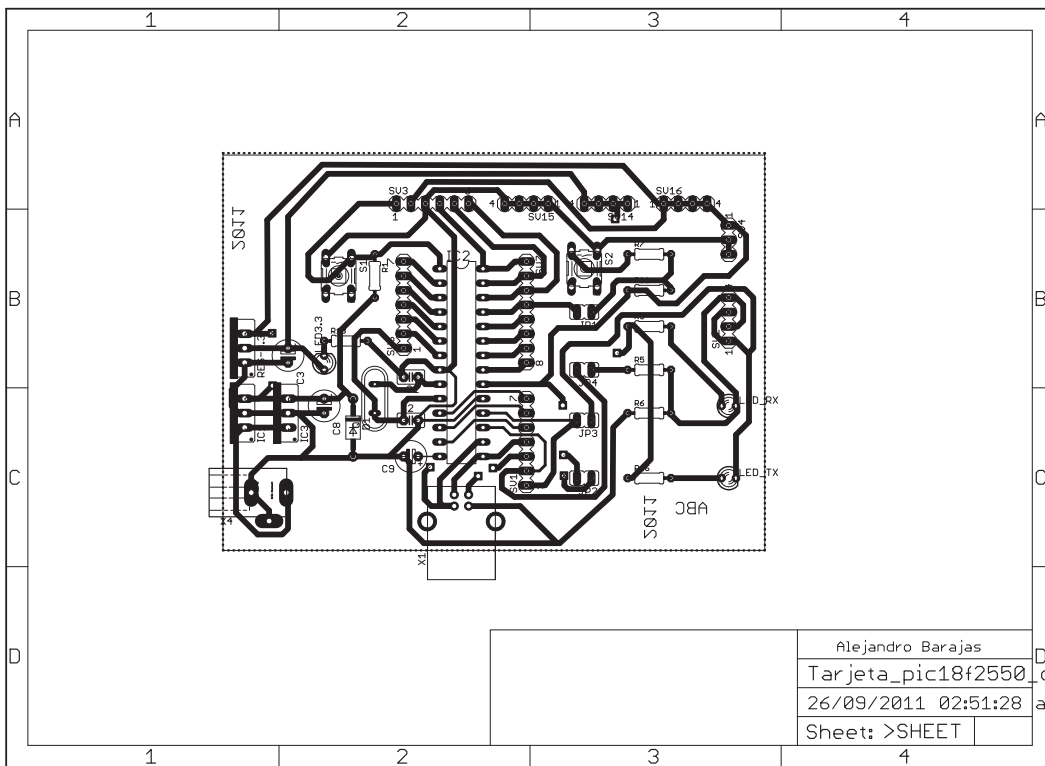
REV: 1.0

Date: 26/09/2011 02:57:47 a.m.

Sheet: 1/1

Apéndice D.10

ESQUEMÁTICOS DE LA TARJETA BASADA EN EL PIC18F2550 ENCARGADA DE
CONTROLAR EL MÓDULO GSM



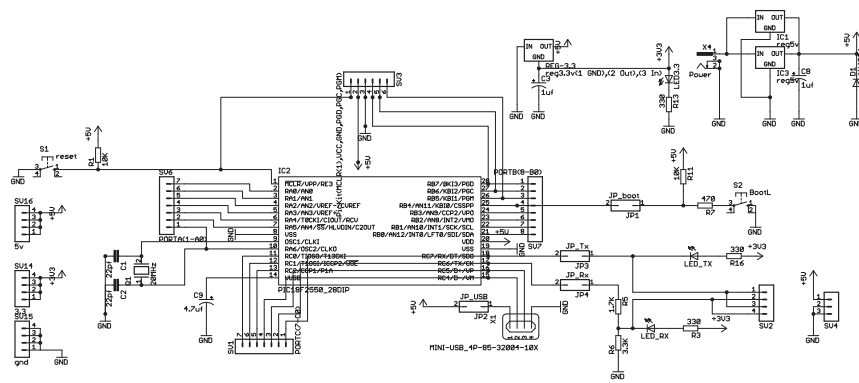
Alejandro Barajas

Tarjeta_pic18f2550

26/09/2011 02:51:28

Sheet: >SHEET

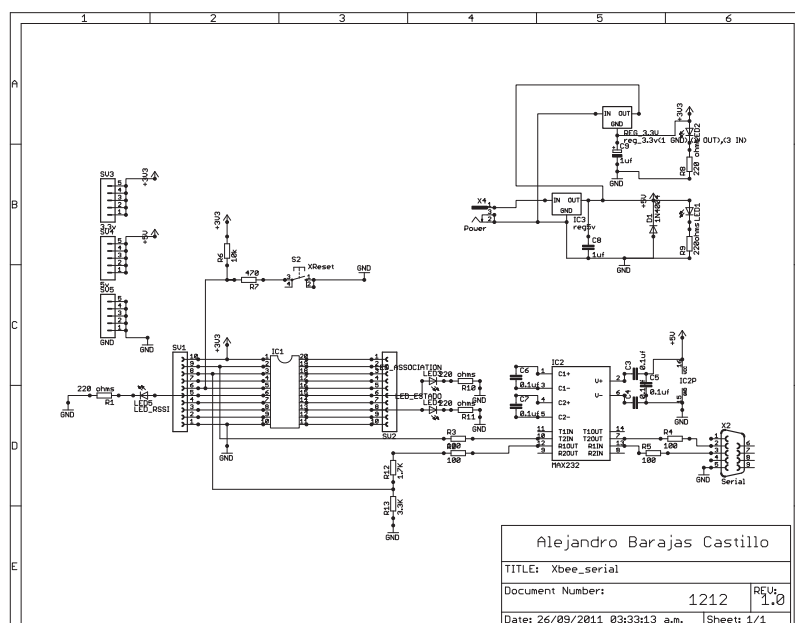
control_modulo_GSM
a.m.



Alejandro Barajas Castillo		
TITLE: Tarjeta_pic18f2550_control_modulo_GSM		
Document Number:	1212	REV 3.0
Date: 26/09/2011 03:35:19 a.m.	Sheet: 1/1	

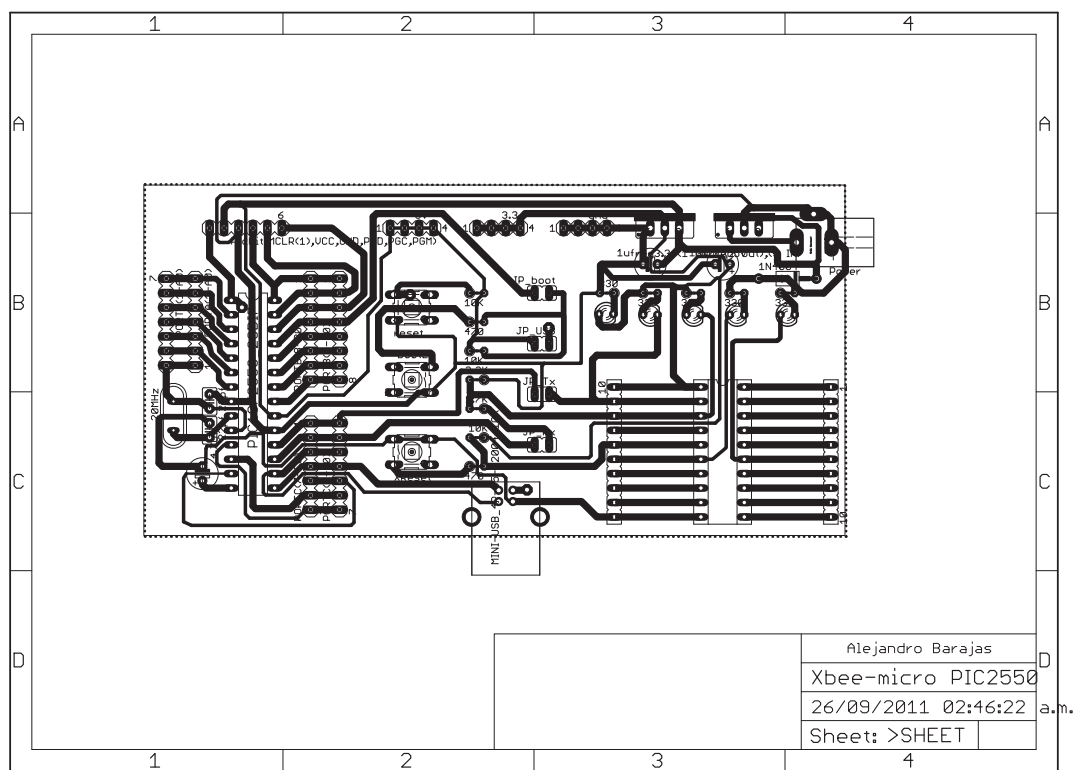
Apéndice D.11

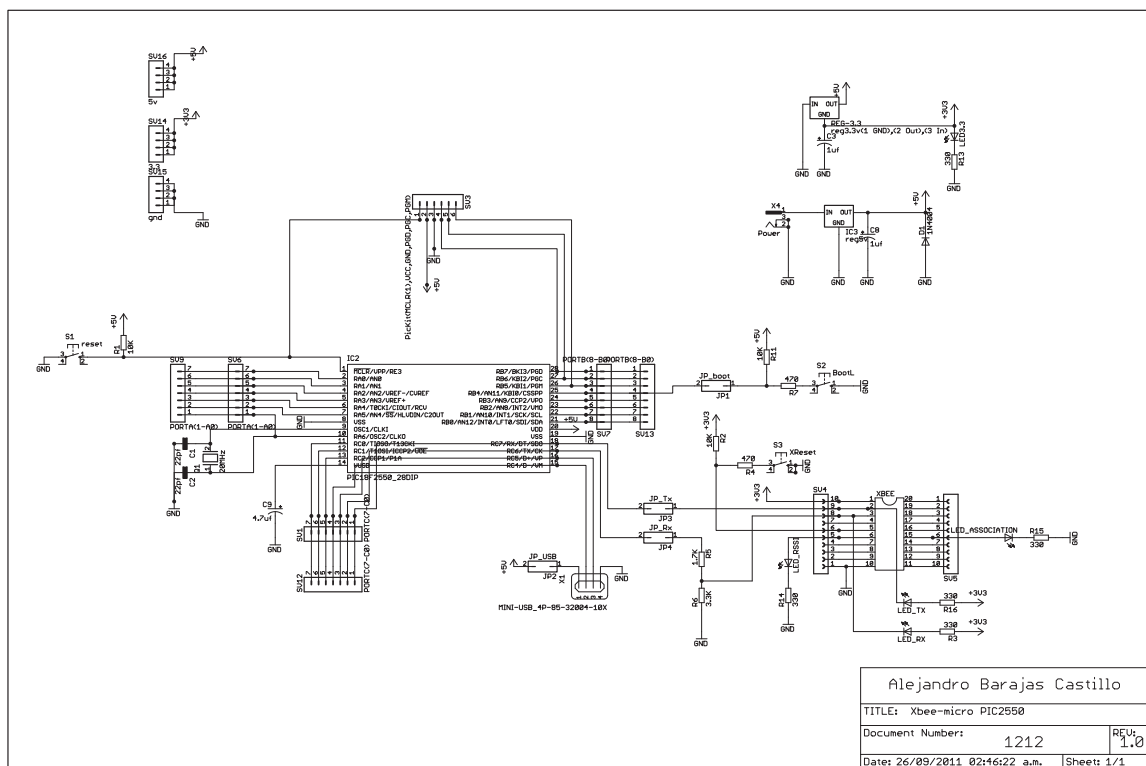
ESQUEMÁTICOS DE LA TARJETA XBEE®-RS232



Apéndice D.12

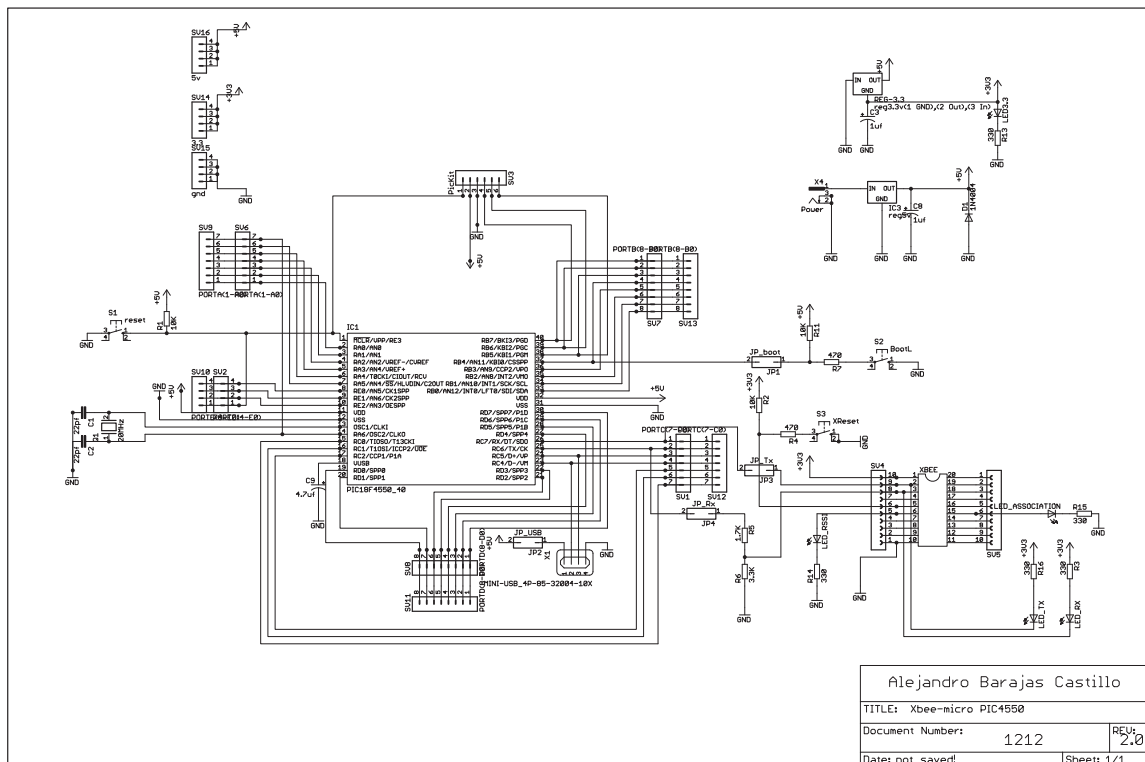
ESQUEMÁTICOS DE LA TARJETA BASADA EN EL PIC18F2550 ENCARGADA DE
CONTROLAR EL MÓDULO XBEE®

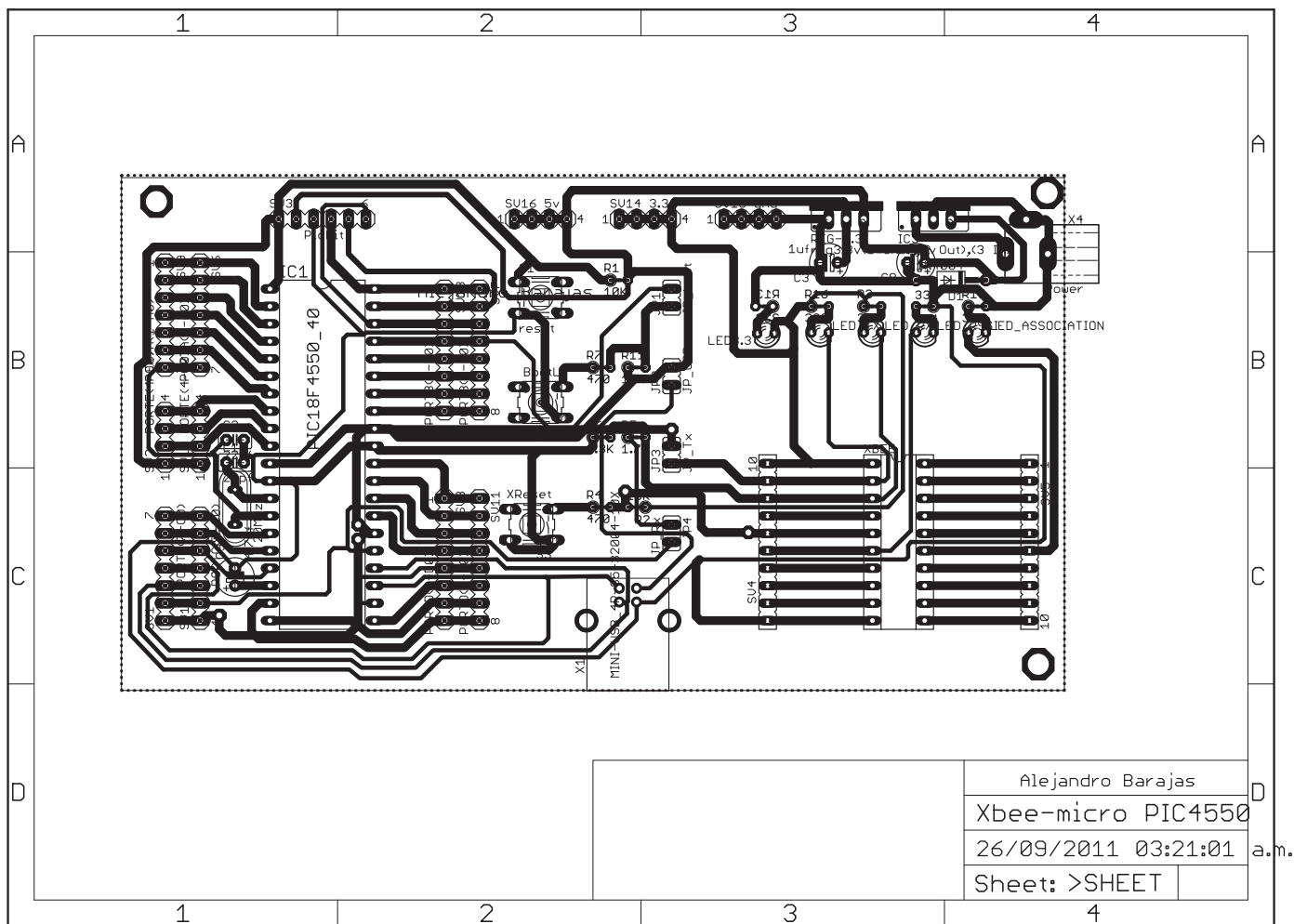




Apéndice D.13

ESQUEMÁTICOS DE LA TARJETA BASADA EN EL PIC18F4550 ENCARGADA DE
CONTROLAR EL MÓDULO XBEE®





Apéndice E

PROGRAMAS PARA EL MANEJO DE LOS ESQUEMAS DE COMUNICACIÓN MEDIANTE

LOS MICROCONTROLADORES

Apéndice E.1

PROGRAMA PARA EL MANEJO DE LOS ESQUEMAS DE RADIOFRECUENCIA E INTERNET

```

//*****//
//          Programa Principal para modificar el sistema
//          mediante internet TCP/IP, o Xbee
//*****//
#include <p18f4550.h>
#include <delays.h>

//*****DEFINIONES DE FUNCIONES*****
unsigned int convierte_canal(unsigned char canal);
void init_ad(void);
void transmite(unsigned char dato);
void timer_isr (void);
void pwm1(unsigned char PWM1L,unsigned char PWM1H);
void pwm2(unsigned char PWM2L,unsigned char PWM2H);
unsigned char recibe(void);

//*****DECLARACION DE VARIABLES*****

unsigned char i=0,dato,dato2,bomba;
unsigned char info=0x30;
unsigned char PWM1H=0,PWM1L=0,PWM2H=0,PWM2L=0;
char output_conv[8],input_pwm[8];
unsigned int valor_ciclo1,ciclo,valor_ciclo2;

/**** VECTOR REMAPPING SOLO PARA BOOTLOADER *****/
extern void _startup(void); //ver c018ic en el directorio del C18
#pragma code _RESET_INTERRUPT_VECTOR = 0x000800 //modifica el vector de reset
void _reset (void)
{
    _asm goto _startup_endasm
}
#pragma code //regresa a la sección de memoria por default

#pragma code _HIGH_INTERRUPT_VECTOR = 0x000808 //modifica el vector de interrupciones alto
void _high_ISR (void)
{
}

#pragma code _LOW_INTERRUPT_VECTOR = 0x000818 //modifica el vector de interrupciones bajo
void _low_ISR (void)
{
    _asm GOTO timer_isr_endasm;
}
#pragma code //regresa a la sección de memoria por default

#pragma interruptlow timer_isr
void timer_isr (void)
{
    if(PIR1bits.TXIF)
    {
        PIR1bits.TXIF=0;
        PIE1bits.TXIE=0; //interrupción por transmisión apagada para transmitir solo 1 dato
    }
} // termina la función de interrupción*/

//*****//
//          Programa Principal
//*****//
void main(void)
{
    //*****Iniciación del puerto B*****
    unsigned char dato;

```

```

LATB = 0;
TRISB = 0;
PORTB = 0;

//*****Iniciación del puerto C*****
PORTC = 0;          //inicializa los bits del CCP como salidas
LATC = 0;
                //TRISC=0b10111001;    // RC1 y RC2 como salidas para el PWM.

                TRISCbits.TRISC6 = 0;                                // configura bits de TX y RX
                TRISCbits.TRISC7 = 1;

                TRISCbits.TRISC1 = 0;                                // configura bits de PWM
                TRISCbits.TRISC2 = 0;
                PORTC = 0x00;

//*****Iniciación del puerto Serial USART*****
TXSTA = 0b00100100;          //Se iniaaliza la transmisión con 8 bits, en modo asíncrono,
                                //sin envío de BREAK y con velocidad de comunicación alta (BRGH = 1).
RCSTA = 0x90;                //Se inicializa la recepción con 8 bits y se configura RC6 y RC7 como pines Tx y Rx
BAUDCON = 0x08;              // Se inicializa la vel. de comunicación con 16 bits (BRG16 = 1)
                                // y se desactiva la auto detección de velocidad de comunicación

                //[SPBRGH:SPBRG]=(Fosc/(4*Vel_Com))-1=(48,000,000/(4*9600))-1= 1249=0x4E1
                SPBRGH = 0x04;          // Vel_Com = Fosc/(4*([SPBRGH:SPBRG]+1))
                SPBRG = 0xE1;           //9600 baudios

//*****interrupciones*****
RCONbits.IPEN = 1;           //permite alta y baja prioridad, activa interrupciones
INTCONbits.GIE = 1;          // interrupciones de alta prioridad
INTCONbits.PEIE = 1;         // interrupciones de baja prioridad

INTCONbits.TMR0IE = 1;       // interrupción EXTERNA POR DESBORDAMIENTO DE TIMER0 ACTIVADA
INTCON2bits.TMR0IP = 0;      // interrupción del Timer0 en baja (0) o alta (1) Prioridad

//*****Interrupción por transmision de datos serial*****
PIE1bits.TXIE=0;            //interrupcion por transmicion apagada
IPR1bits.TXIP=0;            //prioridad baja (0) prioridad alta (1)

//*****CONFIGURACION PWM*****
//*****PWM*****
/* esta funcion convierte el valor del ciclo al que se quiere manejar el pwm para los motores traseros
el periodo del PWM = (159+1)*4*(1/48e6)*16 = 0.21ms
ya que el Ciclo_util es periodo del PWM (solo aqui sera valido el ciclo util de 0% a 100%)
se tiene que:
DCx<9:0> = Ciclo util / ( 4*TOSC x valor_TMR2_Pre-escaler) = 0.21 ms / ( (1/48e6)*16 ) = 640
así el maximo valor que podra tener el Ciclo util sera:
Ciclo util = 640 *(1/48e6)*16 = 0.21 ms
Asi cambiamos para ambos canales el ciclo util que variara deacuerdo al valor recibido deacuerdo a
la baja o alta velocidad a la que se quiera manejar*/
//*****
LATC=0x00;          //Analiza el portc
TMR2=0;            //inicializa el Timer 2

T2CON=0b00000110;   //configura y enciende Timer 2 con pre-escales de 1:16
CCP1CON=0b00001100; //modo PWM con los bits menos significativos del pwm en 0

CCP2CON=0b00001100; //modo PWM con los bits menos significativos del pwm en 0
PR2=255;            //PWM_period=[PR2+1]*4*Tosc(valor_TMR2_Pre-escaler)
CCPR1L=0;           //el periodo esta desde 0x00 a 0xFF
CCPR1H=0;           //canal 1: PWM_Perio=(159+1)*4*(1/48e6)*16=21ms
CCPR2L=0;           //el periodo esta desde 0x00 a 0xFF
CCPR2H=0;           //canal 1: PWM_Perio=(159+1)*4*(1/48e6)*16=21ms
i=0;

```

```
//*****CONFIGURACION CANAL DE CONVERSION ANALOGICO A DIGITAL*****
```

```
init_ad(); //INICIALIZAMOS CONVERTIDOR
RCSTAbits.CREN=0; //recepcion de datos seriales apagada

while(1)
{
    convierte_canal(0);           //convierte canal 0
    //Delay10KTCYx(10);
    output_conv[0]=ADRESH;        //guarda posicion alta canal analogico 0
    output_conv[1]=ADRESL;        //guarda posicion baja canal analogico 0
    convierte_canal(1);
    output_conv[2]=ADRESH;        //guarda posicion alta canal analogico 1
    output_conv[3]=ADRESL;        //guarda posicion baja canal analogico 1
    transmite(0x38);              //bit inicio de datos

    for(i=0;i<=3;i++)
    {
        transmite(output_conv[i]); //transmite conversiones
    }

    RCSTAbits.CREN=1;             //recepcion de datos seriales prendida

    for(i=0;i<=3;i++)
    {
        recibe();
        input_pwm[i]=dato2;       // recibe los datos de labview
    }
    recibe();
    bomba=dato2;                  //recibe estado de la bomba
    if(bomba==0)
    {
        PORTBbits.RB7=1;         //estado de la bomba activado
    }
    else
    {
        PORTBbits.RB7=0;         //estado de la bomba apagado
    }

    RCSTAbits.CREN=0;             //recepcion de datos seriales apagada
    pwm1(input_pwm[0],input_pwm[1]); //modifica el canal PWM 1
    pwm2(input_pwm[2],input_pwm[3]); //modifica el canal PWM 2
}

}
```

```
//*****//
//      Inicialización del convertidor analogico a digital
//*****//
```

```
void init_ad(void)
{
    /*por default en el reset pone el puerto A , B, E como entradas o asegurese
    de que esta en modo AN*/
    ADCON1=0b00001010; //todos los puertos como analogicos y Vref+ a vdd y Vref- a Vss

    ADCON2=0b10000010; /*justifica hacia la derecha un dato de 10 bits se elige la frecuencia focs/32
    tiempo de adquisicion automatica comienza despues de que se india la conversion*/
}
```

```
//*****//
//      rutina para la conversion del A/D
//*****//
```

```
unsigned int convierte_canal(unsigned char canal)
{
    unsigned char ADCON0_RES=0b00111000; //temporal para actualizar canal

    canal=canal<<2; //se posiciona donde esta el numero del canal ADCON0
```

```

ADCON0= canal & 0b00111100; /*actualiza el numero del canal y prende el convertidor*/
/*habilita el ADC*/
ADCON0bits.ADON=1;

canal=0;
Delay100TCYx(3); //25Us espera a que se establezcan los voltajes
ADCON0bits.GO_DONE=1; //inicia conversion
while(ADCON0bits.GO_DONE){}; //ESPERA QUE TERMINE LA CONVERSION
//return(unsigned int)(ADRESH << 8) | ADRESL; //regresa el resultado de la conversion en un entero
return (ADRES);
}

//*****//
//          rutina para modificar el canal PWM 1
//*****//
void pwm1(unsigned char PWM1H,unsigned char PWM1L)
{
    PIR1bits.TMR2IF=0; //limpia bandera del timer 2
    if(PWM1L & 01) //bit 4 del CCP1CON
    {
        CCP1CONbits.DC1B0=1;
    }
    else
    {
        CCP1CONbits.DC1B0=0;
    }

    if(PWM1L & 02) //bit 4 del CCP1CON
    {
        CCP1CONbits.DC1B1=1;
    }
    else
    {
        CCP1CONbits.DC1B1=0;
    }
    ciclo=(char)((PWM1L>>2) | (PWM1H<<6)); //obtiene parte alta de los resultados
    CCPR1L=ciclo;
    return;
}

//*****//
//          rutina para modificar el canal PWM 2
//*****//
void pwm2(unsigned char PWM2H,unsigned char PWM2L)
{
    PIR1bits.TMR2IF=0; //limpia bandera del timer 2
    if(PWM2L & 01) //bit 4 del CCP1CON
    {
        CCP2CONbits.DC2B0=1;
    }
    else
    {
        CCP2CONbits.DC2B0=0;
    }

    if(PWM2L & 02) //bit 4 del CCP1CON
    {
        CCP2CONbits.DC2B1=1;
    }
    else
    {
        CCP2CONbits.DC2B1=0;
    }
    ciclo=(char)((PWM2L>>2) | (PWM2H<<6)); //obtiene parte alta de los resultados
    CCPR2L=ciclo;
    return;
}

```

```

}

//*****//
//      rutina para transmitir datos seriales
//*****//
void transmite(unsigned char dato)
{
    TXREG=dato;
    PIE1bits.TXIE=1; //interrupcion por transmision encendida
    Delay1KTCYx(18);
}

//*****//
//      rutina para recibir datos seriales
//*****//
unsigned char recibe(void)
{
    while(PIR1bits.RCIF==0);
    dato2=RCREG;
}

```

Apéndice E.1.1

FUNCIONES PARA CONFIGURAR LOS MÓDULOS XBEE®

```

void in_at(void);
void at_wr(void);
void at_cn(void);
void atdl(unsigned char dl1,unsigned char dl2,unsigned char dl3,unsigned char dl4,unsigned char dl5,unsigned char dl6,unsigned char
dl7,unsigned char dl8);
void atdh(unsigned char dh1,unsigned char dh2,unsigned char dh3,unsigned char dh4,unsigned char dh5,unsigned char dh6);
void atmy(unsigned char my1,unsigned char my2,unsigned char my3,unsigned char my4);
void atid(unsigned char id1,unsigned char id2,unsigned char id3,unsigned char id4);
void atch(unsigned char ch1,unsigned char ch2);

//*****//
//
//      Función Transmite
//*****//
void transmite(unsigned char dato)
{
//      // transfiere caracter
    TXREG=dato;
    PIE1bits.TXIE=1; //interrupción por transmisión encendida
    Delay1KTCYx(18);
}

//*****//
//
//      Función para entrar a línea de comandos AT
//*****//
void in_at(void)
{
    Delay10KTCYx(240);
    Delay10KTCYx(240);
    Delay10KTCYx(240);
    Delay10KTCYx(240);
    Delay10KTCYx(250);
    transmite(0x2B);
    transmite(0x2B);
    transmite(0x2B);
    Delay10KTCYx(240);
    Delay10KTCYx(240);
    Delay10KTCYx(240);
    Delay10KTCYx(240);
    Delay10KTCYx(250);
    confirmacion();
}
//*****//
//
//      Función que guarda los comandos AT utilizados
//*****//
void at_wr(void)
{
    transmite(0x61);
    transmite(0x74);
    transmite(0x77);
    transmite(0x72);
    transmite(0x0D);
    confirmacion();
}
//*****//
//
//      Función que sale de línea de comandos AT
//*****//
void at_cn(void)
{
    transmite(0x61);
    transmite(0x74);
    transmite(0x63);
    transmite(0x6E);
    transmite(0x0D);
    confirmacion();
}
//*****//
//
//      Función que determina con que dispositivo nos
//      vamos a comunicar
//*****//
void atdl(unsigned char dl1,unsigned char dl2,unsigned char dl3,unsigned char dl4,unsigned char dl5,unsigned char dl6,unsigned char
dl7,unsigned char dl8)

```

```

{

    transmite(0x61);
    transmite(0x74);
    transmite(0x64);
    transmite(0x6C);
    transmite(dl1);
    transmite(dl2);
    transmite(dl3);
    transmite(dl4);
    transmite(dl5);
    transmite(dl6);
    transmite(dl7);
    transmite(dl8);
    transmite(0x0D);
    confirmacion();
}
//*****//
//          Función para determinar la dirección del dispositivo
//*****//
void atmy(unsigned char my1,unsigned char my2,unsigned char my3,unsigned char my4)
{
    transmite(0x61);
    transmite(0x74);
    transmite(0x6D);
    transmite(0x79);
    transmite(my1);
    transmite(my2);
    transmite(my3);
    transmite(my4);
    transmite(0x0D);
    confirmacion();
}
//*****//
//          Función para determinar la red en la que estamos trabajando
//*****//
void atid(unsigned char id1,unsigned char id2,unsigned char id3,unsigned char id4)
{
    transmite(0x61);
    transmite(0x74);
    transmite(0x69);
    transmite(0x64);
    transmite(id1);
    transmite(id2);
    transmite(id3);
    transmite(id4);
    transmite(0x0D);
    confirmacion();
}
//*****//
//          Función para determinar el ancho de banda de la
//          frecuencia en la que se va a trabajar
//*****//
void atch(unsigned char ch1,unsigned char ch2)
{
    transmite(0x61);
    transmite(0x74);
    transmite(0x63);
    transmite(0x68);
    transmite(ch1);
    transmite(ch2);
    transmite(0x0D);
    confirmacion();
}
}

```

Apéndice E.2

PROGRAMA PARA EL MANEJO DEL ESQUEMA GSM

```

//*****
//          Programa Principal para modificar el sistema
//          mediante mensajes de texto con el módulo GSM
//*****
#include <p18f4550.h>
#include <delays.h>

void transmite(unsigned char dato);
void timer_isr (void);

void confirmacion(void);
void borrar(void);

void at(void);
void ini_llamada(void);
void ter_llamada(void);
void sband(unsigned char banda);
void cmgf(void); //INICIALIZACION PARA MANDAR MENSAJES.
void cnmi(void); //INICIALIZACION PARA RECIBIR MENSAJES DE TEXTO.
void cmgs(void); //MENSAJE DE TEXTO
void convierte(unsigned int ADC);
void datos_sistema(void);
void sms_in(void);
void id_num(void);
void llamada_in(void);
void pwm(unsigned int g_verdadx,unsigned int g_verdady);

unsigned int convierte_canal(unsigned char canal);
void init_ad(void);

//*****DECLARACION DE VARIABLES
unsigned short long altura;

unsigned int ele1=0,ele2=0,v_pwm1=0,v_pwm2=0;
unsigned int valor_ciclo=0,ciclo=0;
unsigned int ciclo1=0,ciclo2=0;

unsigned char i=0,dato=0,nbi=0,numero_byte,informacion[8],config_inicio[11]={'+', 'S', 'I', 'N', 'D', 0x3A, 0x20, 0x34, 0x0D, 0x0A};
unsigned char info_t=0,info_r=0,ack=1,j=0,confir[7]={0x0D,0x0A,0x4F,0x4B,0x0D,0x0A},sms[4]={'+', '+', '+'};
unsigned char confir_sms[5]={0x0D,0x0A,0x3E,0x20},registro[8];
unsigned char sms_data[55];
unsigned char call_in[28]={0x52,0x49,0x4E,0x47,0x0D,
                                0x0A,0x0D,
                                0x0A,0x2B,0x43,0x4C,0x49,0x50,0x3A,0x20,0x22,
                                '4','4','3','1','0','2','4','4','8','6',0x22};

//
                                '4','4','3','1','9','9','8','8','1','7',0x22}; //call_in[5]={0x52,0x49,0x4E,0x47};

unsigned char numero[11]={'4','4','3','1','9','9','8','8','1','7'};
//unsigned char numero[11]={'4','4','3','2','2','8','3','7','4','4'};
unsigned char aux,x,k;
unsigned char output_buffer[8];
char caracter2[1]={'.'};

/**** VECTOR REMAPPING SOLO PARA BOOTLOADER *****/
extern void _startup(void); //ver c018ic en el directorio del C18
#pragma code _RESET_INTERRUPT_VECTOR = 0x000800 //modifica el vector de reset
void _reset (void)
{
    _asm goto _startup_endasm
}
#pragma code //regresa a la sección de memoria por default

#pragma code _HIGH_INTERRUPT_VECTOR = 0x000808 //modifica el vector de interrupciones alto

```

```

void _high_ISR (void)
{
    _asm GOTO timer_isr _endasm;
}

#pragma code _LOW_INTERRUPT_VECTOR = 0x000818 //modifica el vector de interrupciones bajo
void _low_ISR (void)
{
}
#pragma code //regresa a la sección de memoria por default
#pragma interruptlow timer_isr
void timer_isr (void)
{
    if(PIR1bits.TeXIF)
    {
        PIR1bits.TXIF=0;
        PIE1bits.TXIE=0; //interrupcion por transmision apagada para transmitir solo 1 dato
    }

    if(PIR1bits.RCIF)
    {
        info_r=RCREG;
        RCREG=0;
        PIR1bits.RCIF=0;
    }
}

//*****//
// Reception de mensajes por interrupcion
//*****//
if(info_r!=0 && ack==1) // si llega mensaje del numero adecuado entra
{
    if(info_r!=numero[j])
    {
        j=0;
    }
    if(info_r==numero[j])
    {
        j++;
    }
    if(info_r==numero[j-1] && j==10)
    {
        ack=2;
        j=0;
    }
    info_r=0;
}
if(info_r!=0 && ack==2) //guarda los datos del mensaje
{
    sms_data[j]=info_r;
    if(j==54)
    {
        ack=1;
        j=0;

        if(sms_data[47]>=0x32 || sms_data[47]==0x30)
        {
            PORTBbits.RB7=0; //modifica estado bomba
        }
        else
        {
            PORTBbits.RB7=1;
        }

        ele1=((sms_data[49]-0x30)*10)+(sms_data[50]-0x30);
        if(ele1>=11) // modifica electrovalvula 1
        {
            ele1=10;
        }
    }
}

```

```

    }
    else
    {
        ele1=ele1;
    }
    v_pwm1=(1023*ele1)/10;

    ele2=((sms_data[52]-0x30)*10)+(sms_data[53]-0x30);
    if(ele2>=11)        //modifica electrovalvula 2
    {
        ele2=10;
    }
    else
    {
        ele2=ele2;
    }
    v_pwm2=(1023*ele2)/10;
    pwm(v_pwm1,v_pwm2);
    x=1;
}
j++;
info_r=0; //limpia variables
}

}
// termina la función de interrupción*/

//*****//
//
//      Programa Principal
//*****//
void main(void)
{
//*****Iniciacion de puertos*****//
//PORTC ccp1 Y ccp2 COMO SALIDAS
    PORTC=0;        //analiza los bits del CCp como salidas
    LATC=0;
    TRISCbits.TRISC1 = 0; // CCP1
    TRISCbits.TRISC2 = 0; // CCP2
    TRISCbits.TRISC6 = 0; // configura bits de TX y RX
    TRISCbits.TRISC7 = 1;

    LATB = 0;
    PORTB = 0;
    ADCON1 = 0x0F;        //todos los canales del PORTB analogicos configurados como I/O digitales.
    TRISB = 0x07;        //pin rb0 como entrada
    PORTB = 0x00;        //inicializa el puerto
//*****SERIAL*****/
    TXSTA = 0b00100100;        //Se iniacliza la transmisión con 8 bits, en modo asíncrono,
                                //sin envío de BREAK y con velocidad de comunicación alta (BRGH = 1).
    RCSTA = 0x90;        //Se inicializa la recepción con 8 bits y se configura RC6 y RC7 como pines Tx y Rx
    BAUDCON = 0x08;        // Se inicializa la vel. de comunicación con 16 bits (BRG16 = 1)
                                // y se desactiva la auto detección de velocidad de comunicación

    //([SPBRGH:SPBRG])=(Fosc/(4*Vel_Com))-1=(48,000,000/(4*9600))-1= 1249=0x4E1
    SPBRGH = 0x04;        // Vel_Com = Fosc/(4*([SPBRGH:SPBRG]+1))
    SPBRG = 0xE1;
//*****PWM*****/
    TMR2=0;        //inicializa el Timer 2

    T2CON=0b00000110;        //configura y enciende Timer 2 con pre-escales de 1:16

    CCP1CON=0b00001100;        //modo PWM con los bits menos significativos del pwm en 0
    PR2=255;        //PWM_period=[PR2+1]*4*Tosc(valor_TMR2_Pre-escaler)
    CCPR1L=0;        //el periodo esta desde 0x00 a 0xFF
    CCPR1H=0;

```

```

        CCP2CON=0b00001100;    //MODO PWM CON LOS DOS BITS MENOS SIGNIFICATIVOS DEL PWM EN 0
        CCPR2L=0;
        CCPR2H=0;
        LATB = 0;

/*****
* Habilita interrupciones
*****/
        RCONbits.IPEN = 1;      //permite alta y baja prioridad, activa interrupciones
        INTCONbits.GIE = 1;     // interrupciones de alta prioridad
        INTCONbits.PEIE = 1;    // interrupciones de baja prioridad

        PIE1bits.TXIE=0;        //interrupcion por transmision apagada
        IPR1bits.TXIP=1;        //prioridad baja (0) prioridad alta (1)

        PIE1bits.RCIE=1;        //interrupcion por recepcion encendida
        IPR1bits.RCIP=1;        //prioridad baja (0) prioridad alta (1)

/*****
* Inicializa modulo GSM
*****/
        PORTBbits.RB7=0;
        init_ad();
        Delay10KTCYx(200);
        Delay10KTCYx(200);

        ack=0;

        confirmacion();
        at();                    //modo at
        cmgfb();                 //mensajes
        sband(7);                //banda
        cnmi();                  //mensajes

        info_r=0;                //bandera
        j=0;
        ack=1;
        PORTB=0x00;
        pwm(0,0);                //pwm a 0
        x=0;

        while(1)
        {
        }
}

/*****//
//          Función Transmite
//*****/
void transmite(unsigned char dato)
{
        TXREG=dato;              // transfiere caracter
        PIE1bits.TXIE=1;        //interrupcion por transmision encendida
        Delay1KTCYx(19);
}

/*****//
//          Función que confirma la entrada a comandos AT
//*****/
void confirmacion(void)
{
        j=0;                    //Se limpian las banderas indicadoras
        ack=0;
        while(ack!=1)

```

```

{
    if(info_r!=0)                //Si llega un dato entra en la rutina
    {
        if(info_r!=config_inicio[j])    //Si es diferente al esperado en la posición deseada
        {                                //limpia las variables
            j=0;
            borrar();
        }
        if(info_r==config_inicio[j])    //Si es igual a la posición deseada aumenta el contador
        {                                // para seguir esperando los datos que faltan
            j++;
        }
        if(info_r==config_inicio[j-1] && j==10)    //Si es igual a la ultima posicion y ya se completo
        {
            //el registro activa la bandera para salir de la rutina
            ack=1;
        }
        info_r=0;
    }
}
borrar(); //limpia variables
ack=0;
}

//*****//
//          Función para borrar el arreglo de datos recibido
//*****//
void borrar(void)
{
    for(i=0;i<=numero_byte;i++)
    {
        informacion[i]=0;
    }
}

void at(void)
{
    Delay10KTCYx(200); // Tiempo para que el módulo esté listo
    transmite(0x41);    //Bytes del comando AT
    Delay1KTCYx(20);
    transmite(0x54);
    Delay1KTCYx(20);
    transmite(0x0D);
    Delay10KTCYx(200); //Tiempo de confirmación
    Delay10KTCYx(200);
}

void ini_llamada(void)
{
    Delay1KTCYx(20); //Bytes del comando ATD
    transmite(0x41); //El cual nos permite realizar
    Delay1KTCYx(20);
    transmite(0x54);
    Delay1KTCYx(20);
    transmite(0x44);
    Delay1KTCYx(20);
    for(i=0;i<=9;i++) //Se introduce el número del receptor
    {
        transmite(numero[i]);
        Delay1KTCYx(20);
    }
    transmite(0x0D);
    Delay10KTCYx(200); //Tiempo de confirmación
    Delay10KTCYx(200);
}

```

```

void ter_llamada(void)
{
    Delay1KTCYx(20); //Bytes del comando ATH
    transmite(0x41); //El cual nos permite terminar las llamadas
    Delay1KTCYx(20);
    transmite(0x54);
    Delay1KTCYx(20);
    transmite(0x48);
    Delay1KTCYx(20);
    transmite(0x0D);
    Delay10KTCYx(200);
    Delay10KTCYx(200);
}

void sband(unsigned char banda)
{
    Delay10KTCYx(200); // Tiempo para que el módulo esté listo
    banda=0x30+banda; //Tomamos el valor de la banda de frecuencia
    Delay1KTCYx(20);
    transmite(0x41); //Bytes del comando at+sband
    Delay1KTCYx(20);
    transmite(0x54);
    Delay1KTCYx(20);
    transmite(0x2B);
    Delay1KTCYx(20);
    transmite(0x53);
    Delay1KTCYx(20);
    transmite(0x42);
    Delay1KTCYx(20);
    transmite(0x41);
    Delay1KTCYx(20);
    transmite(0x4E);
    Delay1KTCYx(20);
    transmite(0x44);
    Delay1KTCYx(20);
    transmite(0x3D);
    Delay1KTCYx(20);
    transmite(banda); //Banda a utilizar
    Delay1KTCYx(20);
    transmite(0x0D);
    Delay10KTCYx(200); //Tiempo de confirmación
    Delay10KTCYx(200);
}

void cmgfv(void)
{
    Delay10KTCYx(200); // Tiempo para que el módulo esté listo
    transmite(0x41); //Bytes del comando at+cmgfv=1 el cual permite
    Delay1KTCYx(20); //seleccionar que modo de mensajes sms usar
    transmite(0x54); //se selecciono el modo mensajes cortos
    Delay1KTCYx(20);
    transmite(0x2B);
    Delay1KTCYx(20);
    transmite(0x43);
    Delay1KTCYx(20);
    transmite(0x4D);
    Delay1KTCYx(20);
    transmite(0x47);
    Delay1KTCYx(20);
    transmite(0x46);
    Delay1KTCYx(20);
    transmite(0x3D);
    Delay1KTCYx(20);
    transmite(0x31);
    Delay1KTCYx(20);
    transmite(0x0D);
    Delay10KTCYx(200);
}

```

```

        Delay10KTCYx(200); //Tiempo de confirmación
    }

void cmgs(void)
{
    Delay10KTCYx(200); // Tiempo para que el módulo esté listo
    transmite(0x41); //Bytes del comando at+cmgs el cual permite transmitir
    Delay1KTCYx(20); //mensajes de texto
    transmite(0x54);
    Delay1KTCYx(20);
    transmite(0x2B);
    Delay1KTCYx(20);
    transmite(0x43);
    Delay1KTCYx(20);
    transmite(0x4D);
    Delay1KTCYx(20);
    transmite(0x47);
    Delay1KTCYx(20);
    transmite(0x53);
    Delay1KTCYx(20);
    transmite(0x3D);

    Delay1KTCYx(20);
    transmite(0x22);

    Delay1KTCYx(20);
    for(i=0;i<=9;i++) //Se introduce el número del receptor
    {
        transmite(numero[i]);
        Delay1KTCYx(20);
    }

    transmite(0x22);
    Delay1KTCYx(20);

    transmite(0x0D);
    //Delay1KTCYx(18);
    Delay1KTCYx(240); //Tiempo de confirmación
    Delay1KTCYx(240);
    Delay1KTCYx(240);

    Delay1KTCYx(20); //Bytes del mensaje a transmitir
    transmite(0x48); //Hola desde el modulo GSM
    Delay1KTCYx(20);
    transmite(0x6F);
    Delay1KTCYx(20);
    transmite(0x6C);
    Delay1KTCYx(20);
    transmite(0x61);
    Delay1KTCYx(20);
    transmite(0x20);
    Delay1KTCYx(20);
    transmite(0x64);
    Delay1KTCYx(20);
    transmite(0x65);
    Delay1KTCYx(20);
    transmite(0x73);
    Delay1KTCYx(20);
    transmite(0x64);
    Delay1KTCYx(20);
    transmite(0x65);
    Delay1KTCYx(20);
    transmite(0x20);
    Delay1KTCYx(20);
    transmite(0x65);
    Delay1KTCYx(20);
}

```

```

transmite(0x6C);
Delay1KTCYx(20);
transmite(0x20);
Delay1KTCYx(20);
transmite(0x6D);
Delay1KTCYx(20);
transmite(0x6F);
Delay1KTCYx(20);
transmite(0x64);
Delay1KTCYx(20);
transmite(0x75);
Delay1KTCYx(20);
transmite(0x6C);
Delay1KTCYx(20);
transmite(0x6F);
Delay1KTCYx(20);
transmite(0x20);
Delay1KTCYx(20);
transmite(0x47);
Delay1KTCYx(20);
transmite(0x53);
Delay1KTCYx(20);
transmite(0x4D);
Delay1KTCYx(20);
transmite(0x0D);
Delay1KTCYx(20);
transmite(0x0A);
Delay1KTCYx(20);

transmite(0x61); //altura tanque 1=
Delay1KTCYx(20);
transmite(0x6C);
Delay1KTCYx(20);
transmite(0x74);
Delay1KTCYx(20);
transmite(0x75);
Delay1KTCYx(20);
transmite(0x72);
Delay1KTCYx(20);
transmite(0x61);
Delay1KTCYx(20);
transmite(0x20);
Delay1KTCYx(20);
transmite(0x74);
Delay1KTCYx(20);
transmite(0x61);
Delay1KTCYx(20);
transmite(0x6E);
Delay1KTCYx(20);
transmite(0x71);
Delay1KTCYx(20);
transmite(0x75);
Delay1KTCYx(20);
transmite(0x65);
Delay1KTCYx(20);
transmite(0x20);
Delay1KTCYx(20);
transmite(0x31);
Delay1KTCYx(20);
transmite(0x3D);
Delay1KTCYx(20);

convierte_canal(0); //Transmisión de la altura 1
convierte(ADRES);

transmite(0x20); //cm

```

```

Delay1KTCYx(20);
transmite(0x63);
Delay1KTCYx(20);
transmite(0x6D);
Delay1KTCYx(20);

transmite(0x0D);
Delay1KTCYx(20);
transmite(0x61);
Delay1KTCYx(20); //altura tanque 1=
transmite(0x6C);
Delay1KTCYx(20);
transmite(0x74);
Delay1KTCYx(20);
transmite(0x75);
Delay1KTCYx(20);
transmite(0x72);
Delay1KTCYx(20);
transmite(0x61);
Delay1KTCYx(20);
transmite(0x20);
Delay1KTCYx(20);
transmite(0x74);
Delay1KTCYx(20);
transmite(0x61);
Delay1KTCYx(20);
transmite(0x6E);
Delay1KTCYx(20);
transmite(0x71);
Delay1KTCYx(20);
transmite(0x75);
Delay1KTCYx(20);
transmite(0x65);
Delay1KTCYx(20);
transmite(0x20);
Delay1KTCYx(20);
transmite(0x32);
Delay1KTCYx(20);
transmite(0x3D);
Delay1KTCYx(20);

convierte_canal(1); //Transmisión de la altura 2
convierte(ADRES);

transmite(0x20); //cm
Delay1KTCYx(20);
transmite(0x63);
Delay1KTCYx(20);
transmite(0x6D);
Delay1KTCYx(20);

Delay1KTCYx(20);
Delay1KTCYx(20); //Tiempo de espera
Delay1KTCYx(20);

transmite(0x1A); //Finalizacion del mensaje
//Delay1KTCYx(18);

Delay1KTCYx(240); //Tiempo de confirmación
Delay1KTCYx(240);
Delay1KTCYx(240);
Delay1KTCYx(240);
Delay1KTCYx(240);
}

```

```

void cnmi(void)
{
    Delay10KTCYx(200); // Tiempo para que el módulo esté listo
    transmite(0x41);    // Bytes del comando at+cnmi el cual permite recibir
    Delay1KTCYx(20);   // mensajes directamente en el módulo
    transmite(0x54);
    Delay1KTCYx(20);
    transmite(0x2B);
    Delay1KTCYx(20);
    transmite(0x43);
    Delay1KTCYx(20);
    transmite(0x4E);
    Delay1KTCYx(20);
    transmite(0x4D);
    Delay1KTCYx(20);
    transmite(0x49);
    Delay1KTCYx(20);
    transmite(0x3D);
    Delay1KTCYx(20);
    transmite(0x33);
    Delay1KTCYx(20);
    transmite(0x2C);
    Delay1KTCYx(20);
    transmite(0x33);
    Delay1KTCYx(20);
    transmite(0x2C);
    Delay1KTCYx(20);
    transmite(0x30);
    Delay1KTCYx(20);
    transmite(0x2C);
    Delay1KTCYx(20);
    transmite(0x30);
    Delay1KTCYx(20);
    transmite(0x0D);
    Delay1KTCYx(240); // Tiempo de confirmación
    Delay1KTCYx(240);
}

void confirmacion_sms(void)
{
    j=0; // Se limpian las banderas indicadoras
    ack=0;
    while(ack!=1)
    {
        if(info_r!=0) // Si llega un dato entra en la rutina
        {
            if(info_r!=confir_sms[j]) // Si es diferente al esperado en la posición deseada
            { // Limpia las variables
                j=0;
                borrar();
            }
            if(info_r==confir_sms[j]) // Si es igual a la posición deseada aumenta el contador
            { // para seguir esperando los datos que faltan
                j++;
            }
            if(info_r==confir_sms[j-1] && j==4) // Si es igual a la última posición y ya se completo
            { // el
                registro activa la bandera para salir de la rutina
                ack=1;
            }
            info_r=0;
        }
    }
    borrar(); // Limpia variables
    ack=0;    // Limpia bandera
}

```

```

//*****//
//      Inicialización del convertidor analogico a digital
//*****//

void init_ad(void)
{
/*por default en el reset pone el puerto A , B, E como entradas o asegurese
de que esta en modo AN*/
    ADCON1=0b00001010; //todos los puertos como analogicos y Vref+ a vdd y Vref- a Vss

    ADCON2=0b10000010; /*justifica hacia la derecha un dato de 10 bits se elige la frecuencia focs/32
tiempo de adquisicion automatica comienza despues de que se india la conversion*/
}

//*****//
//      rutina para la conversion del A/D
//*****//

unsigned int convierte_canal(unsigned char canal)
{
    unsigned char ADCON0_RES=0b00111000; //temporal para actualizar canal

    canal=canal<<2; //se posiciona donde esta el numero del canal ADCON0
    ADCON0= canal & 0b00111000; /*actualiza el numero del canal y prende el convertidor*/
    /*habilita el ADC*/
    ADCON0bits.ADON=1;

    canal=0;
    Delay100TCYx(3); //25Us espera a que se establezcan los voltajes
    ADCON0bits.GO_DONE=1; //inicia conversion
    while(ADCON0bits.GO_DONE){}; //ESPERA QUE TERMINE LA CONVERSION
    //return(unsigned int)(ADRESH << 8) | ADRESL; //regresa el resultado de la conversion en un entero
    return (ADRES);
}

//*****//
//      rutina para modificar el voltaje del canal en centimetros
//*****//
void convierte(unsigned int ADC)
{
    altura=ADC;
    altura=((560*altura)/1023)+10;
    transmite((altura/100+0x30));
    altura=altura%100;
    transmite((altura/10+0x30));
    transmite(caracter2[0]);
    transmite((altura%10+0x30));
}

/*****PWM*****/
/* esta funcion convierte el valor del ciclo al que se quiere manejar el pwm para los motores traseros
el periodo del PWM = (159+1)*4*(1/48e6)*16 = 0.21ms
ya que el Ciclo_util es periodo del PWM (solo aqui sera valido el ciclo util de 0% a 100%)
se tiene que:
DCx<9:0> = Ciclo util / ( 4*TOSC x valor_TMR2_Pre-escaler) = 0.21 ms / ( (1/48e6)*16 ) = 640
así el maximo valor que podra tener el Ciclo util sera:
Ciclo util = 640 *(1/48e6)*16 = 0.21 ms
Asi cambiamos para ambos canales el ciclo util que variara deacuerdo al valor recibido deacuerdo a
la baja o alta velocidad a la que se quiera manejar*/
/*****//
//      rutina para modificar los canales PWM 1 y PWM 2
//*****//

```

```

void pwm(unsigned int g_verdadx,unsigned int g_verdady)
{
    PIR1bits.TMR2IF=0;    //limpia bandera del timer 2
    if(g_verdadx & 01) //bit 4 del CCP1CON
    {
        CCP1CONbits.DC1B0=1;
    }
    else
    {
        CCP1CONbits.DC1B0=0;
    }

    if(g_verdady & 01) //bit 4 del CCP2CON
    {
        CCP2CONbits.DC2B0=1;
    }
    else
    {
        CCP2CONbits.DC2B0=0;
    }

    if(g_verdadx & 02) //bit 4 del CCP1CON
    {
        CCP1CONbits.DC1B1=1;
    }
    else
    {
        CCP1CONbits.DC1B1=0;
    }
    if(g_verdady & 02) //bit 4 del CCP2CON
    {
        CCP2CONbits.DC2B1=1;
    }
    else
    {
        CCP2CONbits.DC2B1=0;
    }
    ciclo1=(char)(g_verdadx>>2);    //obtiene parte alta de los resultados
    ciclo2=(char)(g_verdady>>2);    //obtiene parte alta de los resultados
    CCPR1L=ciclo1;                  //actualiza salida PWM coloca los 8 bits mas significativos
    CCPR2L=ciclo2;                  //actualiza salida PWM coloca los 8 bits mas significativos
}

```

Bibliografía

- [Tomasi, 2003] Wayne Tomasi. “Sistemas de comunicaciones electrónicas”. Prentice Hall. 2003.
- [Microchip, 2009] Microchip Technology Inc. Página principal, Estados Unidos, 10 de enero de 2009, <http://www.microchip.com>
- [Ogata, 1993] Katsuhiko Ogata, Ingeniería de Control Moderna, Prentice Hall 1993.
- [Ramírez, 1998] MC. Salvador Ramírez Zavala, “Diseño y construcción de un manipulador con dos grados de libertad y orientación del efector final”, Tesis de Maestría Octubre de 1998, Universidad Michoacana de San Nicolás de Hidalgo.
- [Robodacta, 2009] Robodacta, Pagina principal, México, 4 de agosto de 2011, <http://www.robodacta.com.mx/>
- [Buitrago, 2006] Julián Alberto Buitrago Giraldo. Evolución de los sistemas móviles celulares GSM. Tesis de maestría escuela de ingeniería eléctrica y electrónica área de telecomunicaciones. Santiago de Cali 2006.
- [Alipso, 2000] Alipso, “protocolos de comunicaciones”, México, 12 de abril de 2000, <http://www.alipso.com/monografias/protocolos/>
- [Aie, 2006] Aie. “Protocolos de comunicaciones industriales”.<http://www.aie.cl/files/file/comites/ca/articulos/agosto-06.pdf>
- [Lasso, 2005] Lasso. “Protocolos de Comunicación Industrial a través de Fibra Óptica”. <http://www.lasso.com.mx/LinkClick.aspx?fileticket=gsjVYEIwAn8%3D&tabid=216>

- [Bautista, 2005] Eduardo Bautista Hernández, “Simulación de los Algoritmos Dijkstra y Bellman-Ford para ruteo de paquetes en redes de comunicaciones”. Tesis de Licenciatura. Mayo del 2005, Departamento de Ingeniería Electrónica, Universidad de las Américas Puebla.
- [Cser, 2006] Felipe Andrés Cser Aedo, “Sistema de control mediante mensajes SMS y Microcontroladores”. Tesis de Licenciatura. Mayo del 2006, Facultad de Ingeniería de la Universidad Católica de Temuco.
- [Palominos, 2006] Néstor Manuel Palominos González “Osciloscopio portátil USB con conexión de red para monitoreo remoto” Tesis de Licenciatura Octubre del 2008, Facultad de Ciencias Físicas y Matemáticas Universidad de Chile.
- [Rosario, 2010] Luis Ángel Rosario Díaz “Diseño de un sistema de propósito general para la gestión de recursos remotos mediante el Sistema Global para las Comunicaciones Móviles” Tesis de Licenciatura Junio del 2010, Universidad Tecnológica de la Mixteca.
- [Manual Xbee® , 2007] El manual del producto de los módulos Xbee® y Xbee-PRO® de MaxStream. <http://www.sparkfun.com/datasheets/Wireless/Zigbee/XBee-Manual.pdf>
- [Características Xbee® , 2007] Características del producto de los módulos Xbee® y Xbee-PRO® de MaxStream. <http://www.sparkfun.com/datasheets/Wireless/Zigbee/XBee-Manual.pdf>
- [Sparkfun, 2011] Sparkfun, Pagina principal, USA, 4 de agosto de 2011. <http://www.sparkfun.com/datasheets/Wireless/Zigbee>
- [Firmware Xbee® , 2010] Procedimiento para actualizar el firmware de los Xbee® y Xbee-PRO® de MaxStream. http://itp.nyu.edu/~raf275/meshnetworking/XBee/XBee_firmware_upgrade.html
- [Digi XBee® , 2011] Página del fabricante (Digi, antes MaxStream). <http://www.digi.com/products/wireless/point-multipoint/xbee-series1-module.jsp>

- [Cadena, 2005] B. Cadena V.M., “Construcción de un módulo para el control de nivel de líquidos”, XIX Jornadas en Ingeniería Eléctrica y Electrónica, Vol. 19, 2005.
- [Gutiérrez,2006] J. Gutiérrez, M.A. Porta-Gándara, “Medidor ultrasónico de nivel de agua para estanques”, Ingeniería Investigación y Tecnología, Vol. VII-4, 2006.
- [Barajas, 2010] A. Barajas “Sintonización e Implementación de un Controlador PID Digital en un Sistema de Nivel de Líquido” XIII Reunión de Otoño de Potencia, Electrónica y Computación ROPEC. Manzanillo Colima Noviembre del 2010.
- [Anzures, 2008] Juan Anzures Marín “Sintonización e Implementación de un Controlador PID Digital en un Sistema de Nivel de Líquido”. Encuentro de Investigación en Ingeniería Eléctrica. Zacatecas, Zacatecas, 13 de marzo del 2008.
- [Manual SM5100B, 2006] El manual del producto del módulo SM5100B. <http://www.sparkfun.com/datasheets/CellularShield/SM5100B%20Datasheet.pdf>
- [Comandos AT del módulo SM5100B, 2008] Comandos AT del módulo SM5100B.<http://www.sparkfun.com/datasheets/CellularShield/SM5100B%20AT%20Command%20Set.pdf>
- [Dreamfebric 2006] SMS and the PDU format.<http://www.dreamfebric.com/sms>
- [Boylestad, 2002] Boylestad “Teoría de circuitos y dispositivos electrónicos”. Prentice Hall. 2003.