



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

UNIDAD DE GESTIÓN DE  TECNOLOGÍAS

**DEPARTAMENTO DE ELECTRÓNICA Y
COMPUTACIÓN.**

**CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN
& AVIÓNICA.**

**PROYECTO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL
TÍTULO DE TECNÓLOGO EN ELECTRÓNICA MENCIÓN
INSTRUMENTACIÓN & AVIÓNICA.**

**TEMA: “IMPLEMENTACIÓN DE UN CONTROL INALÁMBRICO
PARA EL MÓDULO DE ELECTRONEUMÁTICA MEDIANTE
XBEE PRO”.**

**AUTOR: MONTALUISA MONTALUISA, WASHINGTON
MARCELO.**

DIRECTOR: ING. ESPINOSA BRAVO, JESSY JIMENA.

LATACUNGA.

2016



**DEPARTAMENTO DE Y COMPUTACIÓN
CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN &
AVIÓNICA**

CERTIFICACIÓN

Certifico que el trabajo de titulación, **“IMPLEMENTACIÓN DE UN CONTROL INALÁMBRICO PARA EL MÓDULO DE ELECTRONEUMÁTICA MEDIANTE XBEE PRO”** realizado por el señor **MONTALUISA MONTALUISA WASHINGTON MARCELO**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar al señor **MONTALUISA MONTALUISA WASHINGTON MARCELO** para que lo sustente públicamente.

Latacunga, 22 de julio 2016

**SRA. ING. JESSY ESPINOSA
DIRECTOR**



**DEPARTAMENTO DE ELECTRÓNICA Y COMPUTACIÓN
CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN &
AVIÓNICA**

AUTORÍA DE RESPONSABILIDAD

Yo, **MONTALUISA MONTALUISA WASHINGTON MARCELO**, con cédula de identidad N° 0502885981 declaro que este trabajo de titulación **“IMPLEMENTACIÓN DE UN CONTROL INALÁMBRICO PARA EL MÓDULO DE ELECTRONEUMÁTICA MEDIANTE XBEE PRO”** ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

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

Latacunga, 22 de julio de 2016

Montaluisa Montaluisa Washington Marcelo

C.I: 0502885981



**DEPARTAMENTO DE ELECTRÓNICA Y COMPUTACIÓN
CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN &
AVIÓNICA**

AUTORIZACIÓN (PUBLICACIÓN BIBLIOTECA VIRTUAL)

Yo, **MONTALUISA MONTALUISA WASHINGTON MARCELO**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación **“IMPLEMENTACIÓN DE UN CONTROL INALÁMBRICO PARA EL MÓDULO DE ELECTRONEUMÁTICA MEDIANTE XBEE PRO”** cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

Latacunga, 22 de julio de 2016

Montaluisa Montaluisa Washington Marcelo

C.I: 0502885981

DEDICATORIA

Al creador de todas las cosas, el que me ha dado la fortaleza para continuar cuando a punto de caer he estado, por fortalecer mi corazón e iluminar mi mente por ello, con toda la humildad dedico primeramente mi trabajo a Dios.

A mis padres Jorge Montaluisa y Elvia Montaluisa por ser el pilar fundamental en todo lo que soy, por brindarme sus consejos y su apoyo incondicional durante mi formación como ser humano y estudiante.

Mi hermano, Alex Darío por estar conmigo y apoyarme siempre, los quiero mucho.

A mi esposa Valeria Flores, que con su amor me ayuda crecer, que admiro por la valentía de su corazón, su humildad y mi hija Denisse Salome, con sus dulces gestos de cariño me inspira a seguir luchando a diario

Gracias de todo corazón.

AGRADECIMIENTO

A Dios por darme la fortaleza y la fe para cumplir muy importante en mi vida. A mi familia, por su apoyo incondicional en todo en que los necesite, siendo mi inspiración, fortaleza y ejemplo de superación para culminar mi carrera profesional. A mi tutora la Ing. Jessy Espinosa por toda su ayuda y por su tiempo invertido para la corrección de mi Proyecto Técnico, al Ing. Pablo Pilatasig por su valiosa orientación y apoyo para el desarrollo de la mi Proyecto Técnico, quienes desde la formación académica siempre estuvieron guiándome y compartiendo los conocimientos.

Muchas gracias y que Dios los bendiga.

ÍNDICE DE CONTENIDOS

CERTIFICACIÓN.....	ii
AUTORÍA DE RESPONSABILIDAD	iii
AUTORIZACIÓN (PUBLICACIÓN BIBLIOTECA VIRTUAL)	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
ÍNDICE DE CONTENIDOS.....	vii
ÍNDICE DE TABLAS	xi
ÍNDICE DE FIGURAS.....	xii
RESUMEN.....	xv
ABSTRACT.....	xvi
CAPÍTULO I.....	1
PLANTEAMIENTO DEL PROBLEMA.....	1
1.1 ANTECEDENTES.....	1
1.2 PLANTEAMIENTO DEL PROBLEMA.....	1
1.3 JUSTIFICACIÓN.....	2
1.4 OBJETIVOS.....	2
1.4.1 Objetivo general.....	2
1.4.2 Objetivos específicos.....	2
1.5 ALCANCE.....	3
CAPÍTULO II.....	4
MARCO TEÓRICO.....	4
2.1 Control.....	4
2.2 ZigBee	5
2.2.1 El estándar ZigBee.....	5
2.2.2 Tipos de tráfico.....	6

2.2.3	Tipos de topologías de red.....	7
2.2.4	Clasificación de redes	9
2.2.5	Aplicaciones de ZigBee.....	10
2.2.6	Futuro del ZigBee.....	10
2.3	XBee.....	11
2.3.1	XBEE PRO (SERIE 1).....	11
2.3.2	Características Técnicas Xbee Pro	12
2.3.3	Descripción de lo pines del módulo Xbee PRO s1	13
2.4	Tarjeta de programación para el módulo XBee-PRO s1.....	14
2.5	Arduino	14
2.5.1	Hardware.....	15
2.5.2	Software Arduino.....	16
2.5.3	Ventajas	16
2.6	Arduino Mega	17
2.6.1	Memoria	19
2.6.2	Entrada y salida.....	19
2.6.3	Comunicación	19
2.7	Arduino uno	19
2.7.1	Memoria	21
2.7.2	Entrada y salida.....	21
2.7.3	Comunicación	21
2.8	Software Proteus	21
2.8.1	Módulos del software Proteus.....	21
2.9	Electroneumática	22
2.9.1	Válvula Neumática	22
2.9.2	Electroválvula.....	23
2.9.3	Tipos de electroválvula.....	23

2.9.4 Cilindro neumático.....	26
2.10 UNIDAD DE MANTENIMIENTO.....	27
CAPÍTULO III.....	29
DESARROLLO DEL TEMA.	29
3.1 Preliminares.....	29
3.2 Requerimientos mínimos.....	29
3.3 Instalación del software X-CTU	29
3.4 Configuración del módulo XBee pro mediante el software x-ctu.....	33
3.4.1 Secciones principales del software XCTU.....	34
3.4.2 INSTALAR DIGI_USE_RF_Drivers	37
3.4.3 CONFIGURAR LOS MÓDULOS XBEE PRO S1	37
3.5 Configuración Arduino mega	47
3.6 Configuración Arduino uno	50
3.7 Código de programación del control Inalámbrico.....	53
3.7.1 Aplicación manual	53
3.7.2 Aplicación automática:	58
3.8 Circuito del control inalambrico.....	67
3.9 Esquema del circuito de potencia en Proteus.....	68
3.10 Esquema del shield del arduino mega.....	69
3.11 Esquema del shield del arduino uno.....	70
3.12 Esquemas de conexión del Modulo Electroneumatica	71
3.13 Pruebas de funcionamiento.....	72
3.13.1 Monitor serie del arduino mega con la aplicación manual.....	72
3.13.2 Pruebas en el módulo de electroneumática	74
CAPÍTULO IV.....	77
CONCLUSIONES Y RECOMENDACIONES.....	77
4.1 Conclusiones.	77

4.2 Recomendaciones.....	78
GLOSARIO DE TÉRMINOS	79
REFERENCIAS BIBLIOGRAFICAS.....	80

ÍNDICE DE TABLAS

Tabla 1	9
Tabla 2	13
Tabla 3	18
Tabla 4	20
Tabla 5	40
Tabla 6	41

ÍNDICE DE FIGURAS

Figura 1 Configuración tipo estrella.....	7
Figura 2 Configuración tipo malla.....	8
Figura 3 Configuración tipo árbol.	8
Figura 4 Redes Inalámbricas.	9
Figura 5 Xbee Pro s1.	11
Figura 6 Pines del módulo Xbee Pro S1.	13
Figura 7 Pines del módulo Xbee Pro S1.	14
Figura 8 Pines del módulo Xbee Pro S1.	15
Figura 9 Software arduino.....	16
Figura 10 Arduino Mega.....	17
Figura 11 Partes del arduino mega.	18
Figura 12 Arduino uno.....	19
Figura 13 Partes del Arduino uno.....	20
Figura 14 Software Proteus	21
Figura 15 Válvula de cajón neumática	23
Figura 16 Electroválvula CPE18-M1H-3GL-1/4.....	23
Figura 17 Válvula 2/2.	24
Figura 18 Válvula 3/2.	24
Figura 19 Válvula 4/2.	24
Figura 20 Válvula 5/2.	25
Figura 21 Válvula de 3 posiciones.	25
Figura 22 Cilindro simple efecto.....	26
Figura 23 Cilindro doble efecto.	27
Figura 24 Unidad de mantenimiento.	27
Figura 25 Pantalla DIGI XCTU.	30
Figura 26 Activacion de licencia.....	30
Figura 27 Selección de Sistema Opertivo.	31
Figura 28 Direccion del software XCTU.	31
Figura 29 Listo para instalar software.	32
Figura 30 Instalando software.	32
Figura 31 Instalacion completada.	33
Figura 32 Logo del software XCTU.	33

Figura 33 Pantalla del software XCTU.....	34
Figura 34 Barra de menus.	34
Figura 35 Barra principal de herramientas.	35
Figura 36 Iconos para agragar modulos.	35
Figura 37 Funcionalidad XCTU.....	35
Figura 38 Modos de trabajo XCTU.....	35
Figura 39 Modulos de radio locales.	36
Figura 40 Area de trabajo.	37
Figura 41 Seleccionar puerto serial.....	38
Figura 42 Paramatros Xbee maestro con MAC: 0013A200406E285F.....	38
Figura 43 Parametros Xbee esclavo con MAC 0013A200406AEF16.	39
Figura 44 Parametros modificados Xbee maestro con MAC 0013A200406E285F	40
Figura 45 Parametros modificados Xbee Pro esclavo con MAC: 0013A200406AEF16.....	42
Figura 46 Cambio a modo consola del XCTU.....	43
Figura 47 Consola de XCTU.	43
Figura 48 Control inalambrico Xbee Pro maestro con MAC 0013A200406E285F.....	44
Figura 49 Control inalambrico Xbee Pro esclavo con MAC: 0013A200406AEF16.....	44
Figura 50 Tx de datos del Xbee Maestro con MAC 0013A200406E285F...	45
Figura 51 Rx de datos del Xbee esclavo con MAC: 0013A200406AEF16..	45
Figura 52 Tx de datos del Xbee esclavo con MAC: 0013A200406AEF16. .	46
Figura 53 Rx de datos del Xbee Maestro con MAC 0013A200406E285F. .	46
Figura 54 Conexión USB entre la tarjeta Arduino Mega y la PC.....	47
Figura 55 Codigo transmisor en la plataforma Arduino.	48
Figura 56 Selección placa arduino mega.	48
Figura 57 Puerto serial arduino mega.	49
Figura 58 Cargar el programa a la placa arduino mega.....	49
Figura 59 Conexión USB entre la tarjeta Arduino Uno y la PC.	50
Figura 60 Codigo del RX en la plataforma arduino.	51
Figura 61 Selecion de la tarjeta arduino uno.....	51
Figura 62 Puerto serial del arduino uno.	52

Figura 63 Subiendo el Programa a la placa arduino uno.	52
Figura 64 Circuito de Control (Maestro).	67
Figura 65 Circuito de Potencia (Esclavo).	67
Figura 66 Esquema de potencia.	68
Figura 67 Circuito Diseñado para el esquema de potencia.....	68
Figura 68 Shield arduino mega.	69
Figura 69 Esquema diseñado del shield arduino mega.	69
Figura 70 Shield arduino uno.	70
Figura 71 Esquema diseñado del shield arduino uno.	70
Figura 72 Esquema Neumatico.....	71
Figura 73 Esquema Electrico.	71
Figura 74 VP, C1, C2, C3 apagados.....	72
Figura 75 VP activado, C1, C2, C3 apagados.....	72
Figura 76 VP, CA activados y CB, CC apagados.....	73
Figura 77 VP, C1, C2, activados y C3 apagado.....	73
Figura 78 VP, C1, C2, C3 activadas.	74
Figura 79 VP, C1, C2, C3 apagados.....	74
Figura 80 VP encendida, C1, C2, C3 adentro.	75
Figura 81 VP encendida, C1 afuera y C2, C3 adentro.	75
Figura 82 VP encendida, C1, C2 afuera y C3 adentro.	76
Figura 83 VP encendida, C1, C2, C3 afuera.....	76

RESUMEN

El presente Proyecto Técnico trata de la implementación de un control con comunicación inalámbrica para el módulo de electroneumática mediante Xbee Pro, basado en las características, ventajas y los campos de aplicación del uso de la tecnología ZigBee ante las demás tecnologías Bluetooth y WiFi, como lo es la reducción de costos de operación, instalación y mantenimiento. Los dispositivos utilizados son XBee-PRO (maestro) con MAC: 0013A200406E285F, el Xbee Pro S1 (esclavo) con MAC: 0013A200406AEF16, Arduino Uno y Arduino Mega. Para la comunicación se utilizó el software XCTU, en el cual se configuró el canal, la potencia, la identificación de red y dirección propia para los módulos Xbee Pro S1. La programación se lo hizo en la plataforma arduino la cual es necesaria para realizar la aplicación en el Arduino Mega (maestro) y el Arduino Uno (esclavo) para interconectar cada uno de los elementos y establecer la comunicación y la visualización en tiempo real del control inalámbrico. Se colocó una pantalla LCD (4x16) conectada en el maestro para visualizar el estado de la válvula principal y los cilindros de doble efecto, además se elaboró los shields Xbee, cuyos esquemáticos se diseñaron en el software Proteus para cada arduino y el circuito de potencia para la activación de las electroválvulas. Dentro de las pruebas de funcionamiento, los módulos trabajaron exitosamente sin interferencias cumpliendo la secuencia automática para los cilindros A+, B+, C+, C-, B-, A-.

Palabras claves

Arduino Mega

Arduino Uno

Electroválvula

Xbee Pro

XCTU

ABSTRACT

This technical project is about the implementation of a control with a Wireless communication for the module of electropneumatic through Xbee Pro. It is based on characteristics, advantages and fields of application of the use of technology ZigBee facing other technologies such as Bluetooth and WiFi, in order to reduce the operation costs, maintenance and installation. The devices used are XBee-Pro S1 (master) with MAC: 0013A200406E285F, Xbee Pro S1 (slave) with 0013A200406AEF16, Arduino Uno and Arduino Mega. For communication was used, XCTU software in which the channel was configured, the potency, the network identification and proper direction for the S1 Pro XBee Pro S1 modules used. The programming was done in Arduino Platform, which is necessary to make the application on the Arduino Mega (master) and Arduino 1 (slave) for interconnecting each one of the elements and for establishing the communication as well as the display in real time of wireless control. An LCD (4x16) screen was placed and it was hooked up with the master for visualizing the state of the principal valve and the Cylinders of dual function. In addition, the Xbee shields were made, their schematic were designed in the Proteus software for each arduino and the potency circuit for activation of the electrovalves. Among the performance tests, the modules worked successfully without interferences, conveying the automatic sequence for the cylinders A+, B+, C+, C-, B-, A-.

KEY WORDS:

Arduino Mega

Arduino Uno

Electrovalve

Xbee Pro

XCTU

Checked by: Lic. Elisa Coque

DOCENTE INGLÉS UGT

CAPÍTULO I

PLANTEAMIENTO DEL PROBLEMA.

1.1 ANTECEDENTES.

El laboratorio de Instrumentación Virtual de la carrera de Electrónica debe disponer de equipos de última tecnología para desarrollar prácticas de comunicación inalámbrica, que vayan a la par con las necesidades del ámbito industrial. Para lo cual se hizo una investigación de campo en el cual se pudo observar que se dispone de un sistema de telemetría con XBee –PRO 900 de un trabajo técnico realizado por la Srta. (Andi, 2012) Con el tema “VISUALIZACIÓN SE SEÑALES DE VELOCIDAD Y PRESIÓN DESDE EL MINI UAV MEDIANTE EL XBEE-PRO 900”. El proyecto realizado ayudo a mejorar y profundizar sobre modos de comunicación como la inalámbrica, permitiendo transmitir datos a largas distancias para aficionados de vehículos aéreos no tripulados.

Tomando como referencia esta investigación se pretende utilizar este tipo de modo de comunicación, para envío y recepción de datos a largas distancias de dispositivos remotos como XBee –PRO, fortaleciendo así el proceso de enseñanza-aprendizaje gracias a la ayuda de dispositivos inalámbricos de última tecnología.

1.2 PLANTEAMIENTO DEL PROBLEMA.

La comunicación inalámbrica es la tecnología más prometedora de esta década que ha sufrido un desarrollo tecnología muy alto siendo productos resistentes de uso industrial y doméstico, ayudando a monitorear y modificar diferentes parámetros a largas distancias donde no puede ser manipulado por la mano humana.

Es indispensable que los estudiantes de la carrera de Electrónica tenga conocimiento de este tipo de tecnología y puedan manipular este tipo de dispositivos remotos en especial el módulo XBee PRO puesto que su aplicabilidad es amplia en aplicaciones de control tales como, automatización de casas, sistemas de seguridad, monitoreo de sistemas remotos, alarmas contra incendio, plantas tratadoras de agua.

De no implementar este tipo de tecnología el personal técnico, que se está preparándose en la Unidad de Gestión y Tecnologías no tendrá los conocimientos necesarios sobre este tipo de tecnología para desenvolverse en el campo laboral.

1.3 JUSTIFICACIÓN.

El presente proyecto de grado se realizará debido a la necesidad de implementar un control inalámbrico para el Modulo de Electroneumática con el fin de familiarizar a los estudiantes de sexto nivel de la carrera de Electrónica el uso y manejo de la plataforma arduino y módulo XBee-PRO facilitando así el desarrollo de prácticas de laboratorio de Instrumentación Virtual.

La comunicación inalámbrica es un excelente medio que permite a docentes y estudiantes integrar diferentes tecnologías con el fin de encontrar la solución a un problema gracias a los módulos XBee que tiene mayor alcance para establecer comunicación entre dispositivos remotos enviando datos confiables a su receptor.

El módulo desarrollado es factible pues se conocería la parte funcional del módulo Xbee-Pro con una distancia máxima de 90m con obstáculos para la transferencia de datos fiables, disponiendo de una plataforma libre de programación Arduino.

1.4 OBJETIVOS.

1.4.1 Objetivo general.

Implementar un control inalámbrico para el módulo de electroneumática mediante XBee-PRO.

1.4.2 Objetivos específicos.

- Investigar las características y especificaciones del XBee-PRO apoyado en manuales técnicos existentes.
- Establecer los requerimientos mínimos de hardware y software para la implementación de control inalámbrico.
- Desarrollar un control inalámbrico mediante XBee-PRO y el módulo de electroneumática.

- Realizar pruebas de funcionamiento del control inalámbrico mediante una secuencia básica de actuadores.

1.5 ALCANCE.

Este proyecto está dirigido a la Carrera de Electrónica Mención Instrumentación y Aviónica de la Unidad de Gestión de Tecnologías en especial en el desarrollo de procesos automatizados. Será de gran beneficio para los alumnos de quinto, sexto nivel permitiéndoles realizar prácticas de control de procesos y automatización brindando los conocimientos teóricos y prácticos acordes al ámbito laboral.

CAPÍTULO II

MARCO TEÓRICO.

2.1 Control.

La palabra control “proviene del término francés *contrôle* y significa comprobación, inspección, fiscalización o intervención. También puede hacer referencia al dominio, mando y preponderancia, o a la regulación sobre un sistema”. (Jara, 2008)

Según Cabrera, 2012 la palabra control:

Tiene muchas connotaciones y su significado depende de la función o del área en que se aplique; puede ser entendida:

- 1.- Como la función administrativa que hace parte del proceso administrativo junto con la planeación, organización y dirección, y lo que la precede.
- 2.- Como los medios de regulación utilizados por un individuo o empresa, como determinadas tareas reguladoras que un controlador aplica en una empresa para acompañar y avalar su desempeño y orientar las decisiones. También hay casos en que la palabra control sirve para diseñar un sistema automático que mantenga un grado constante de flujo o de funcionamiento del sistema total; es el caso del proceso de control de las refinerías de petróleo o de industrias químicas de procesamiento continuo y automático: el mecanismo de control detecta cualquier desvío de los patrones normales, haciendo posible la debida regulación.

También hay otras connotaciones para la palabra control:

- 1.- Comprobar o verificar;
- 2.- Regular;
- 3.- Comparar con un patrón;
- 4.- Ejercer autoridad sobre alguien (dirigir o mandar);
- 5.- Frenar o impedir.

Evidentemente todas esas definiciones representan concepciones incompletas del control, quizás definidas en un modo subjetivo y de aplicación; en definitiva, debe entenderse el control como:

Una función administrativa, ya que conforma parte del proceso de administración, que permite verificar, constatar, palpar, medir, si la actividad, proceso, unidad, elemento o sistema seleccionado está cumpliendo y/o alcanzando o no los resultados que se esperan.

2.2 ZigBee

Es un protocolo de comunicaciones inalámbricas basado en el estándar IEEE 802.15.4 y su función es la de solucionar los problemas de interoperabilidad, duración de la batería y costos de los protocolos propietarios en las aplicaciones de domótica. (Londoño, 2013)

Según Dignani, 2011 se encontró lo siguiente:

ZigBee es un estándar que define un conjunto de protocolos para el armado de redes inalámbricas de corta distancia y baja velocidad de datos. Opera en las bandas de 868 MHz, 915 MHz y 2.4 GHz y puede transferir datos hasta 250Kbps. Esta organización sin fines de lucro nace en el año 2002. Desarrolla un protocolo que adopta al estándar IEEE 802.15.4 para sus 2 primeras capas, es decir la capa física (PHY) y la subcapa de acceso al medio (MAC) y agrega la capa de red y de aplicación. La Capa Física puede trabajar en uno de tres rangos de frecuencias: 868 MHz, 915 MHz o 2.4 GHz con velocidades de hasta 20 Kbps, 40 Kbps y 250 Kbps y cuya utilización de bandas de frecuencias es Europa, JUL y el resto del mundo, respectivamente. ZigBee puede funcionar con tan bajo consumo de energía que las baterías pueden durar años, más de mil veces que en otras aplicaciones inalámbricas.

2.2.1 El estándar ZigBee

Según Dignani, 2011 Estándar ZigBee es:

- Ultra bajo consumo que permita usar equipos a batería
- Bajo costo de dispositivos y de instalación y mantenimiento de ellos.
- Optimizado para ciclo efectivo de transmisión menor a 0.1 %
- Velocidad de transmisión menor que 250 kbps. Típica: menor que 20 kbps

- Tasa de transmisión de hasta 250 kbps en 2.4 GHz, 40kbps en 915MHz y 20 kbps en 868 MHz.
- Usa CSMA-CA (Carrier Sense Multiple Access Collision Avoidance) para acceso al canal.
- Produce alto rendimiento y baja latencia para dispositivos de bajo ciclo de trabajo, muy adecuado esto para sensores y controles.
- 64 bits de direccionamiento determina una cantidad máxima de $1.8 \cdot 10^{19}$ dispositivos.
- 16 bits para identificar redes que determina un total de 65536 redes.
- Permite el uso de ranuras de tiempo (time slots) para posibilitar aplicaciones de baja latencia.
- Protocolo con handshake (diálogo) para mejorar la seguridad en las transferencias.
- Rango: hasta 50 m (valor típico, depende del ambiente). (Dignani, 2011)

2.2.2 Tipos de tráfico

2.2.2.1 Datos periódicos (continuo)

La aplicación define una tasa de datos. Es un caso típico de sensores en donde por ejemplo un sensor necesita transmitir la temperatura cada 10 segundos. (Dignani, 2011)

2.2.2.2 Datos intermitentes (por eventos)

Según el estudio de Dignani, 2011 encontró lo siguiente:

En este caso la aplicación junto a otros estímulos externos al dispositivo definen la tasa de datos. Por ejemplo en un sistema domótico, los interruptores de luces transmiten solo ante un cambio de posición. Mientras tanto están desconectados (comúnmente denominado en modo dormir) y consumiendo una energía de batería mínima. (Dignani, 2011)

2.2.2.3 Datos periódicos con comunicación garantizada (GTS) (Guaranteed time slot)

Según el estudio de Dignani, 2011 encontró lo siguiente:

Hay aplicaciones de baja latencia que requieren comunicación libre de competencia por el canal. GTS es un método de calidad de servicio que garantiza la atención por un cierto Δt dentro de un período T llamado Supertrama. IEEE 802.15.4 provee un modo de trabajo denominado “con baliza” que sirve como multiplexación temporal. (Dignani, 2011)

2.2.3 Tipos de topologías de red

2.2.3.1 Tipo Estrella

Según Ecuared, 2015 Tipo Estrella es:

En la topología en estrella, todos los dispositivos en la red sólo pueden comunicarse con el coordinador PAN. Un caso típico en la formación de una red en estrella es que un dispositivo FDD programado para ser un coordinador PAN se activa y comienza a establecer su red. Lo primero que hace el coordinador PAN es seleccionar un identificador único el cual no es utilizado por cualquier otra red dentro del radio de cobertura de ese dispositivo, en otras palabras, se asegura que el identificador no sea utilizado por cualquier otra red cercana. (Ecuared, 2015)

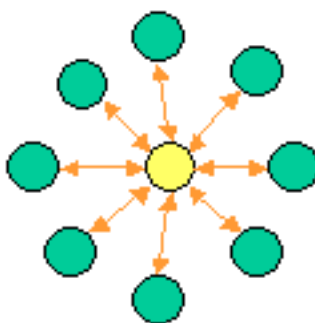


Figura 1 Configuración tipo estrella.

Fuente: (Teloco, 2013)

2.2.3.2 Tipo Malla.

Una red punto a punto puede tomar diferente formas mediante la definición de las restricciones de los dispositivos que pueden comunicarse entre sí. Si

no hay ninguna restricción, la red punto a punto es conocida como una topología de malla. (Ecuared, 2015).

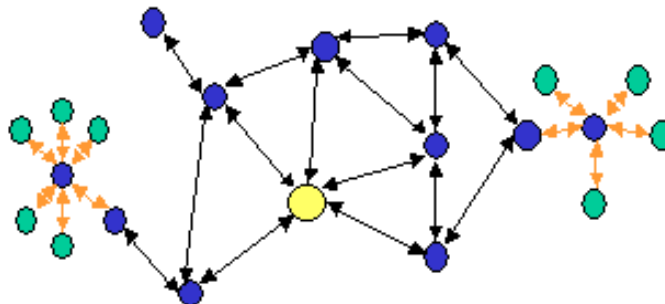


Figura 2 Configuración tipo malla.

Fuente: (Teloco, 2013)

2.2.3.3 Tipo Racimo de Árbol:

Según Ecuared, 2015 encontró lo siguiente:

Otra forma de red punto a punto que ZigBee soporta es una topología de árbol. En este caso, un coordinador de ZigBee (coordinador del PAN), establece la red inicial. Los routers ZigBee son los encargados de formar las ramas y transmitir los mensajes. Los dispositivos finales ZigBee actúan como hojas del árbol y no participan en el enrutamiento de mensajes. La topología de Árbol ayuda a transmitir el mensaje alrededor de la barrera y poder llegar dispositivo. Esto se conoce como salto múltiple (multihopping) debido a que un mensaje realiza saltos de un nodo a otro hasta llegar a su destino. (Ecuared, 2015).

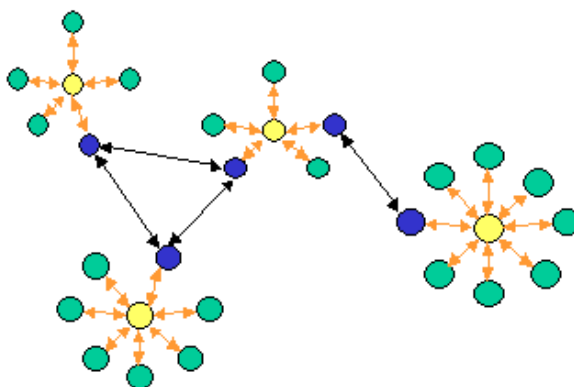


Figura 3 Configuración tipo árbol.

Fuente: (Teloco, 2013)

2.2.4 Clasificación de redes

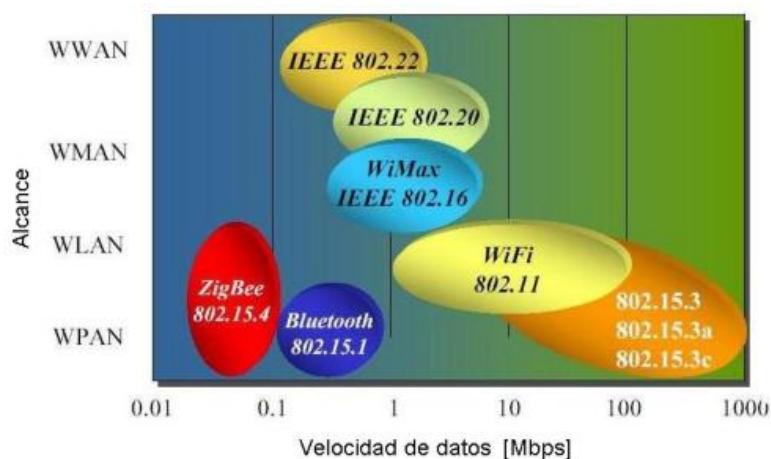


Figura 4 Redes Inalámbricas.

Fuente: (Dignani, 2011)

De acuerdo a su cobertura las redes inalámbricas se clasifican:

- WPAN: Redes inalámbricas de área personal.
- WLAN: Redes inalámbricas de área local.
- WMAN: Redes inalámbricas de área metropolitana.
- WWAN: Redes inalámbricas de área geográfica. (Dignani, 2011).

Tabla 1

Descripción de los pines del módulo Xbee Pro S1.

Prestaciones	ZigBee 802.15.4	Bluetooth 802.15.1	WiFi 802.11
Costo del chip (USD)	1	3	4
Consumo de corriente (mA)	30	65-170	350
Capacidad de red (nodos)	65000	30	7
Vida útil de la batería (días)	>365	7	1
Velocidad de transmisión RF (Kbps)	250	1000-3000	5400
Potencia de transmisión (mW)	1-2	1-100	40-200
Frecuencia de radio (Ghz)	0,968; 0.915; 2.4	2.4	2.4
Rango de trabajo (m)	1-100	1-10	30-100

Fuente: (Zambrano, 2015)

2.2.5 Aplicaciones de ZigBee

El estudio de Dignani, 2011 se encontró lo siguiente:

Automatización en el hogar

Es una de las aplicaciones más usadas de ZigBee ya que es muy fácil la instalación de dispositivos y la modificación de posición de los mismos. Los usos típicos son:

Seguridad

Sensores de movimiento, de rotura de cristales, apertura de puertas y ventanas. A pesar de su baja velocidad también se usa para transmitir imágenes de cámara de seguridad de baja calidad.

Lectura de instrumentos de servicios

Los medidores de consumo de agua, gas y energía eléctrica deben leerse en forma regular a efecto de facturar los servicios. Es posible crear una red tipo malla para que la información de los medidores llegue directamente a la empresa de servicio. También los medidores ZigBee podrían comunicarse con los artefactos dentro de la casa.

Sistema de riego automático

El uso de un medidor de humedad de suelo permite mejorar la eficiencia del consumo de agua. Se puede distribuir una red de sensores de humedad en un parque de modo que solo se riegue las zonas secas y controlar el tiempo de regado. Una red inalámbrica de sensores facilita enormemente la instalación y el mantenimiento. (Dignani, 2011)

2.2.6 Futuro del ZigBee

Según Glen, 2012 el futuro del ZigBee:

Se espera que los módulos ZigBee sean los transmisores inalámbricos más baratos de la historia, y además producidos de forma masiva y dispondrán de una antena integrada, control de frecuencia y una pequeña batería. Ofrecerán una solución tan económica porque la radio se puede fabricar con muchos menos circuitos analógicos de los que se necesitan habitualmente. (Glen, 2012)

2.3 XBee

Los módulos XBee son dispositivos que integran un transmisor - receptor de ZigBee y un procesador en un mismo módulo, lo que le permite a los usuarios desarrollar aplicaciones de manera rápida y sencilla. (Jimenez, 2011)

2.3.1 XBEE PRO (SERIE 1)



Figura 5 Xbee Pro s1.

Fuente: (Xbee, 2015)

Según Duarte, 2014 Xbee Pro S1 es:

Los módulos XBee utilizan el protocolo IEEE 802.15.4 mejor conocido como ZigBee. El protocolo de comunicaciones inalámbrico basado en el estándar de comunicaciones para redes inalámbricas IEEE 802.15.4. Creado por ZigBee Alliance, una organización, teóricamente sin ánimo de lucro, de más de 200 grandes empresas (destacan Mitsubishi, Honeywell, Philips, ODEM do, Invensys, entre otras), muchas de ellas fabricantes de semiconductores. ZigBee permite que dispositivos electrónicos de bajo consumo puedan realizar sus comunicaciones inalámbricas. Es especialmente útil para redes de sensores en entornos industriales, médicos y, sobre todo, doméstico. En especialmente útil para redes de sensores en entornos industriales, médicos y, sobre todo, domésticas. Fueron diseñados para aplicaciones que requieren de un alto tráfico de datos, baja latencia y una sincronización de comunicación predecible. Por lo que básicamente XBee es propiedad de Digi basado en el protocolo ZigBee. En términos simples, los XBee son módulos inalámbricos fáciles de usar. Este es el muy popular módulo Xbee XBP24-AWI-001 Serie 1 de 2.4GHz

fabricado por Digi (formalmente Maxstream). La serie Pro tiene el mismo pin_out y set de instrucciones que la serie básica más un incremento de la potencia de salida a 60mW!. Estos módulos utilizan el stack 802.15.4 (la base de ZigBee) y lo empaquetan en un set de instrucciones simples de utilizar. El Xbee Pro permite una comunicación muy simple y confiable entre microcontroladores, computadores, sistemas y realmente cualquier dispositivo con un puerto serial, Redes punto a punto y multipunto pueden ser configuradas utilizando este equipo. (Duerte, 2014)

2.3.2 Características Técnicas Xbee Pro

Según Xbee, 2015 Características Técnicas Xbee Pro son:

- Corriente pico durante Tx (250 mA).
- Rx Corriente 3.3v; 55 mA.
- 60mW de salida (+18dBm).
- Antena PCB.
- 6 pines de entrada de 10-bit ADC.
- 8 pines digitales de Entrada/Salida.
- Encriptación en 128 bit.
- Dimensiones: 27 mm x 33 mm x 9 mm.
- Soporta conexiones de redes punto a punto y multi-punto.
- Conexión punto a punto, punto a multipunto.
- La velocidad de transmisión de datos de una red ZigBee es de hasta 256kbps.
- Buen Alcance: hasta 1 milla (1.6 Km) en línea de vista y 300 ft (90 m) con obstáculos para los módulos Xbee Pro.
- Bajo consumo <50mA cuando están en funcionamiento y <10uA cuando están en modo sleep.
- Interfaz serial. (Xbee, 2015)

2.3.3 Descripción de los pines del módulo Xbee PRO S1

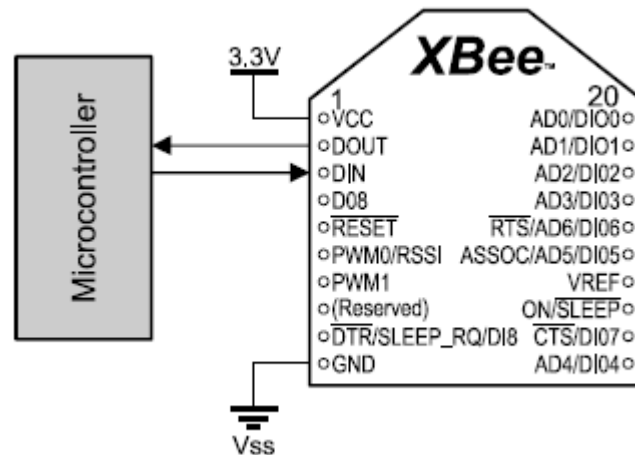


Figura 6 Pines del módulo Xbee Pro S1.

Fuente: (Tymkers, 2015)

Tabla 2

Descripción de los pines del módulo Xbee Pro S1.

PIN	NOMBRE	DESCRIPCION	PIN	NOMBRE	DESCRIPCION
1	Vcc	Power Out	11	Ad4	Input analógico 4
2	Dout	Uart data on	12	Cts	Flujo de control
3	Dint/config	Uart data in	13	On Sleep	Indicador de estatus
4	Dos	Digital output	14	Vref	Voltaje referencial
5	Reset	Módulo reset	15	Ad5	Input analógico 5/digital i/o5
6	Pwm0	Pwm0	16	Rts/Ad6	Input analógico 6/digital i/o6
7	Pwm1	Pwm1	17	Ad3	Input analógico 3/digital i/o3
8	Reservado	Vacio	18	Ad2	Input analógico 2/digital i/o2
9	Dtr/sleep	Control sleep	19	Ad1	Input analógico 1/digital i/o1
10	Gnd	Tierra	20	Ad0	Input analógico 0/digital i/o0

Fuente: (Zambrano, 2015)

2.4 Tarjeta de programación para el módulo XBee-PRO s1

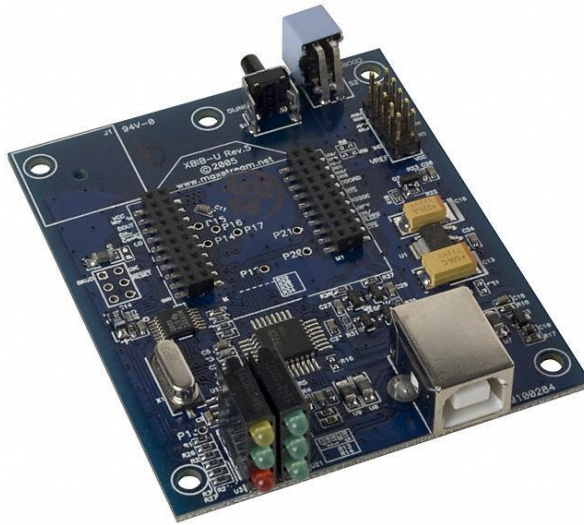


Figura 7 Pines del módulo Xbee Pro S1.

Fuente: (Digikey, 2016)

Según Digikey, 2016 se encontró lo siguiente:

Es una placa fácil de usar, necesita de un cable USB para la comunicación con la PC. En la parte superior se observa los pines en donde se coloca el módulo Xbee Pro s1 y se utiliza para la configuración de diferentes parámetros del módulo mencionado con la ayuda del software XCT-U. (Digikey, 2016)

2.5 Arduino

Según Weebly, 2015 Arduino es:

Arduino es una plataforma de prototipos electrónica de código abierto (open – source) basada en hardware y software flexibles y fáciles de usar. Está pensado e inspirado en artistas, diseñadores, y estudiantes de computación o robótica y para cualquier interesado en crear objetos o entornos interactivo, o simplemente por hobby. Arduino consta de una placa principal de componentes eléctricos, donde se encuentran conectados los controladores principales que gestionan los demás complementos y circuitos ensamblados en la misma.

Arduino también simplifica el proceso de trabajo con micro controladores, ya que está fabricada de tal manera que viene “pre ensamblada” y lista con los controladores necesarios para poder operar

con ella una vez que la saquemos de su caja, ofreciendo una ventaja muy grande para profesores, estudiantes y aficionados interesados en el desarrollo de tecnologías. Las posibilidades de realizar proyectos basados en esta plataforma tienen como limite la imaginación de quien opera esta herramienta. (Weebly, 2015)

2.5.1 Hardware



Figura 8 Pines del módulo Xbee Pro S1.

Fuente: (Atmels, 2016)

Según Weebly, 2015 Hardware es:

Arduino está constituido en el hardware por un micro controlador principal llamado Atmel AVR de 8 bits (que es programable con un lenguaje de alto nivel), presente en la mayoría de los modelos de Arduino, encargado de realizar los procesos lógicos y matemáticos dentro de la placa, además de controlar y gestionar los recursos de cada uno de los componentes externos conectados a la misma. Consta además de una amplia variedad de sensores eléctricos como cámaras VGA, sensores de sonido, seguidores de línea, botones de control de sensores, e incluso, otras placas de micro controladores (mejor conocidos como Shields), que pueden adaptarse fácilmente gracias a que Arduino cuenta con entradas de pines analógicos y digitales para integrar estos componentes sin necesidad de alterar el diseño original de esta placa. Estos a su vez son controlados junto con el procesador primario por otros componentes de menor jerarquía, pero de igual importancia y prioridad, como el Atmega168, Atmega328, Atmega1280 y el Atmega8 , que son lo más utilizados debido a sus bajos precios y gran flexibilidad para construir diversidad de diseños.

2.5.2 Software Arduino

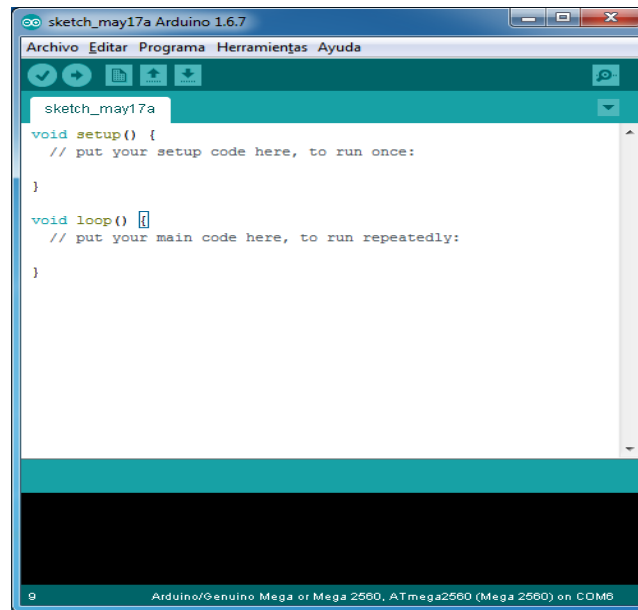


Figura 9 Software arduino.

Fuente: (Arduino, 2016)

Según Weebly, 2015 se encontro lo siguiente:

Como se había mencionado, Arduino, no sólo son componentes eléctricos ni una placa de circuitos, sino que además, también es una plataforma que combina esto con un lenguaje de programación que sirve para controlar los distintos sensores que se encuentran conectados a la placa, por medio de instrucciones y parámetros que nosotros establecemos al conectar la placa a un ordenador. Este lenguaje que opera dentro de Arduino se llama Wirirng, basado en la plataforma Processing y primordialmente en el lenguaje de programación C/C++, que se ha vuelto popular a tal grado de ser el más preferido para enseñar programación a alumnos de nivel superior que estudian computación y robótica, gracias que es muy fácil de aprender y brinda soporte para cualquier necesidad de computación. De este lenguaje derivan otros más que son muy utilizados en el ámbito de Ingeniería y desarrollo, como C#, Java, BASIC, Php, Phytom, JavaScript, Perl, entre otros más. (Weebly, 2015)

2.5.3 Ventajas

El estudio de Weebly, 2015 se encontro lo siguiente:

- **Asequible** - Las placas Arduino son más asequibles comparadas con otras plataformas de microcontroladores.
- **Multi-Plataforma** - El software de Arduino funciona en los sistemas operativos Windows, Macintosh OSX y Linux.
- **Entorno de programación simple y directo** - El entorno de programación de Arduino es fácil de usar para principiantes y lo suficientemente flexible para los usuarios avanzados.
- **Software ampliable y de código abierto**- El software Arduino está publicado bajo una licencia libre y preparado para ser ampliado por programadores experimentados. El lenguaje puede ampliarse a través de librerías de C++.
- **Hardware ampliable y de Código abierto** - Arduino está basado en los microcontroladores ATMEGA168, ATMEGA328 y ATMEGA1280. Los planos de los módulos están publicados bajo licencia Creative Commons, por lo que diseñadores de circuitos con experiencia pueden hacer su propia versión del módulo, ampliándolo u optimizándolo. (Weebly, 2015)

2.6 Arduino Mega



Figura 10 Arduino Mega.

Fuente: (Commerce, 2016)

Según Arduino, 2016 Arduino Mega es

El Arduino Mega es una placa electrónica basada en el ATmega1280. Cuenta con 54 pines digitales de entrada / salida (de los cuales 14 se pueden utilizar como salidas PWM), 16 entradas analógicas, 4 UARTs (puertos serie de hardware), un oscilador de 16MHz, una conexión USB, un conector de alimentación, una cabecera

ICSP, y un botón de reinicio. Contiene todo lo necesario para apoyar el microcontrolador; basta con conectarlo a un ordenador con un cable USB o la corriente con un adaptador de CA a CC o una batería para empezar. (Arduino, Arduino mega, 2016)

Tabla 3

Características técnicas arduino mega

Microcontrolador	ATmega1280
Tensión de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límites)	6-20V
E / S digitales prendedores	54 (de los cuales 15 proporcionan salida PWM)
Pines de entrada analógica	Dieciséis
Corriente continua para Pin I / O	20 Ma
Corriente CC para Pin 3.3V	50 mA
Memoria flash	128 KB de los cuales 4 KB utilizado por el gestor de arranque
SRAM	8 KB
EEPROM	4 KB
Velocidad de reloj	16 MHz

Fuente: (Arduino, Arduino mega, 2016)

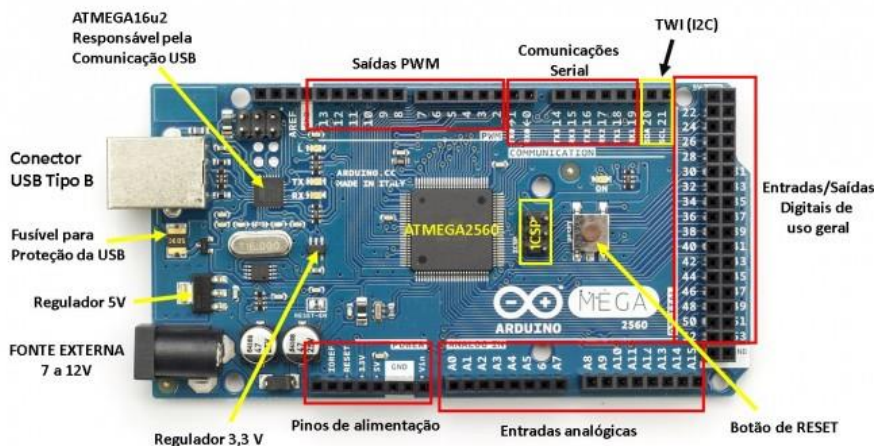


Figura 11 Partes del arduino mega.

Fuente: (Souza, 2014)

2.6.1 Memoria

El estudio de Arduino, Arduino mega, 2016 se encontro lo siguiente:

El ATmega1280 tiene 128 KB de memoria flash para almacenar el código (de los cuales se utiliza 4 KB para el gestor de arranque), 8 KB de SRAM y 4 KB de EEPROM (que puede ser leído y escrito con la librería EEPROM). (Arduino, Arduino mega, 2016)

2.6.2 Entrada y salida

Cada uno de los 54 pines digitales en la Mega se puede utilizar como una entrada o salida y operan a 5 voltios. Cada pin puede proporcionar o recibir un máximo de 40 mA. (Arduino, Arduino mega, 2016)

2.6.3 Comunicación

El Arduino Mega tiene una serie de instalaciones para la comunicación con un ordenador, otro Arduino, u otros microcontroladores y El Atmega1280 ofrece cuatro UART hardware para TTL (5V) de comunicación en serie. (Arduino, Arduino mega, 2016)

2.7 Arduino uno



Figura 12 Arduino uno.

Fuente: (Ariston, 2013)

Según Arduino, 2016 se encontró lo siguiente:

El Uno es una placa electrónica basada en el Atmega328P. Cuenta con 14 pines digitales de entrada / salida (de los cuales 6 se podrán utilizar como salidas PWM), 6 entradas analógicas, un cristal de cuarzo de 16 MHz, una conexión USB, un conector de alimentación, una jefe de ICSP y un botón de reinicio. Contiene todo lo necesario para apoyar el

microcontrolador; basta con conectarlo a un ordenador con un cable USB o la corriente con un adaptador de CA a CC o una batería para empezar. (Arduino, 2016)

Tabla 4

Características Técnicas Arduino Uno

Microcontrolador	<u>Atmega328P</u>
Tensión de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límite)	6-20V
E / S digitales prendedores	14 (de los cuales 6 proporcionan salida PWM)
PWM digital pines I / O	6
Pines de entrada analógica	6
Corriente continua para Pin I / O	20 mA
Corriente CC para Pin 3.3V	50 mA
Memoria flash	32 KB (Atmega328P) de los cuales 0,5 KB utilizado por cargador de arranque
SRAM	2 KB (Atmega328P)
EEPROM	1 KB (Atmega328P)
Velocidad de reloj	16 MHz

Fuente: (Arduino, Arduino Uno, 2016)

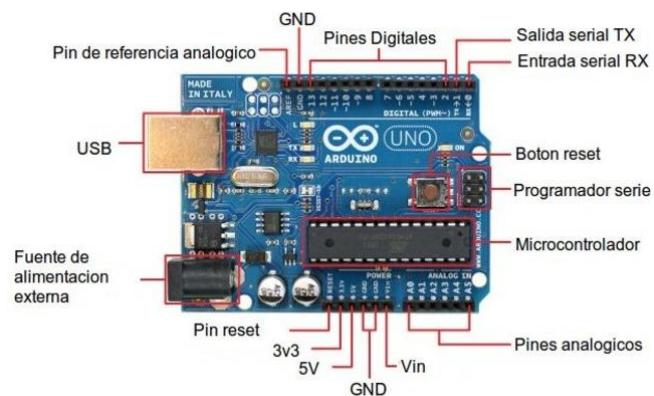


Figura 13 Partes del Arduino uno.

Fuente: (Isaac_PE, 2014)

2.7.1 Memoria

El ATmega328 tiene 32 KB (con 0,5 KB ocupadas por el gestor de arranque). También tiene 2 KB de SRAM y 1 KB de EEPROM. (Arduino, 2016)

2.7.2 Entrada y salida

Cada uno de los 14 pines digitales en el Uno se puede utilizar como una entrada o salida, operan a 5 voltios. (Arduino, 2016)

2.7.3 Comunicación

Según la investigación de Arduino, Arduino uno, 2016 se encontro lo siguiente:

El Uno tiene una serie de instalaciones para la comunicación con un ordenador, la otra tabla Uno, u otros microcontroladores. El ATmega328 ofrece UART TTL (5V) de comunicación en serie, que está disponible en los pines digitales 0 (RX) y 1 (TX). (Arduino, 2016)

2.8 Software Proteus

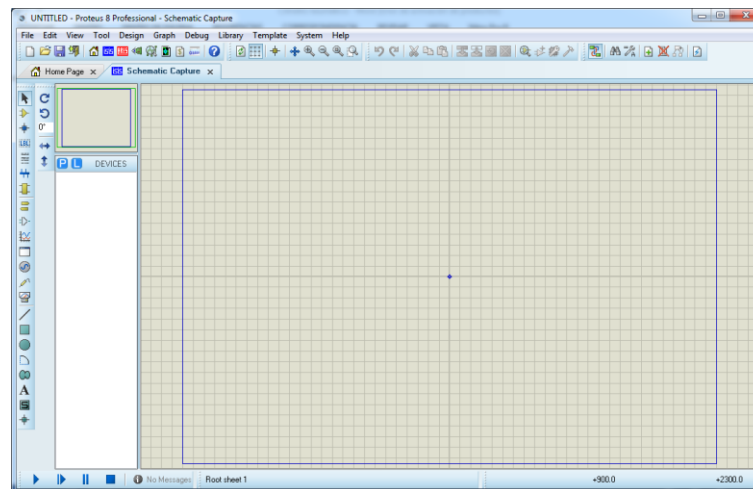


Figura 14 Software Proteus

Fuente: (Proteus V8.1, 2015)

Proteus es un software de simulación para circuitos eléctricos tanto análogos como digitales, además de que te permite crear el layout del PCB y visualizarlo en 3D. (leespain, 2012)

2.8.1 Módulos del software Proteus

El estudio de leespain, 2012 se encontro lo siguiente:

ISIS.- Es el encargado de realizar el modelo esquemático del circuito, para ello cuenta con una librería de más de 6,000 dispositivos tanto analógicos como digitales.

ARES.- Se encarga de hacer la placa de circuito impreso (PCB) además de que puede posicionar automáticamente los componentes y hacer las pistas.

Prospice.- El componente más importante, pues se encarga de simular el comportamiento del circuito.

VSM.- Los que han trabajado con micro-controladores en Proteus sabrán lo útil que es este módulo. Te permite simular el comportamiento de un micro-controlador de las familias PIC, AVR, y otras, sólo le cargas el archivo HEX y Proteus lo simula, además puede interactuar con diferentes periféricos. (leespain, 2012)

2.9 Electroneumática

Según Blogger, 2011 la palabra Electroneumatica es:

En electroneumática, la energía eléctrica substituye a la energía neumática como el elemento natural para la generación y transmisión de las señales de control que se ubican en los sistemas de mando. Los elementos nuevos y/o diferentes que entran en juego están constituidos básicamente para la manipulación y acondicionamiento de las señales de voltaje y corriente que deberán de ser transmitidas a dispositivos de conversión de energía eléctrica a energía neumática para lograr la activación de los actuadores neumáticos. (Blogger, 2011)

2.9.1 Válvula Neumática

Según Parreño, 2012 Valvula Neunatica es:

Una válvula neumática es un elemento de regulación y control de la presión y el caudal del aire a presión. Este aire es recibido directamente después de su generación o sino desde un dispositivo de almacenamiento. Las válvulas dirigen, distribuyen o pueden bloquear el paso del aire para accionar los elementos de trabajo (los actuadores). (Parreño, 2012)



Figura 15 Válvula de cajón neumática

Fuente: (Airtec, 2012)

2.9.2 Electroválvula

Según Altec, 2015 la palabra electroválvula es:

Una electroválvula también conocida como válvula solenoide de uso general es una válvula que abre o cierra el paso de un líquido en un circuito. La apertura y cierre de la válvula se efectúa a través de un campo magnético generado por una bobina en una base fija que atrae el émbolo. (Altec, 2015)



Figura 16 Electroválvula CPE18-M1H-3GL-1/4

Fuente: (Festo, 2015)

2.9.3 Tipos de electroválvula

El estudio de Microautomacion, 2014 encontró lo siguiente:

2.9.3.1 Válvulas 2/2

Pertenece a este grupo todas las válvulas de cierre que poseen un orificio de entrada y otro de salida (2 vías) y dos posiciones de mando. Sólo se utilizan en aquellas partes de los equipos neumáticos donde no es preciso efectuar por la misma válvula la descarga del sistema alimentado; sólo actúan como válvulas de paso. Pueden ser normal

cerradas o normal abiertas, según cierren o habiliten el paso respectivamente en su posición de reposo. (Microautomacion, 2014)

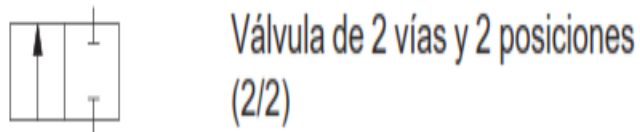


Figura 17 Válvula 2/2.

Fuente: (Microautomacion, 2014)

2.9.3.2 Válvulas 3/2

Además de alimentar a un circuito, permiten su descarga al ser conmutadas. También las hay normalmente cerradas o abiertas. (Microautomacion, 2014)

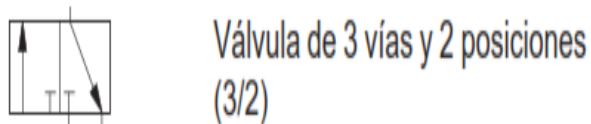


Figura 18 Válvula 3/2.

Fuente: (Microautomacion, 2014)

2.9.3.3 Válvulas 4/2

El estudio de Microautomacion, 2014 encontró lo siguiente:

Poseen cuatro orificios de conexión correspondiendo uno a la alimentación, dos a las utilidades y el restante al escape, el que es común a ambas utilidades. Operan en dos posiciones de mando, para cada una de las cuales sólo una utilización es alimentada, en tanto la otra se encuentra conectada a escape; esta condición se invierte al conmutar la válvula. (Microautomacion, 2014)

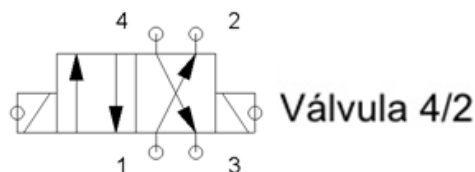


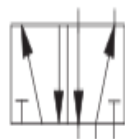
Figura 19 Válvula 4/2.

Fuente: (Microautomacion, 2014)

2.9.3.4 Válvulas 5/2

El estudio de Microautomacion, 2014 encontró lo siguiente:

Éstas poseen cinco orificios de conexión y dos posiciones de mando. A diferencia de la 4/2, poseen dos escapes correspondiendo uno a cada utilización. Esto brinda la posibilidad, entre otras cosas, de controlar la velocidad de avance y retroceso de un cilindro en forma independiente. (Microautomacion, 2014)



Válvula de 5 vías y 2 posiciones
(5/2)

Figura 20 Válvula 5/2.

Fuente: (Microautomacion, 2014)

2.9.3.5 Válvulas de 3 posiciones

El estudio de Microautomacion, 2014 encontró lo siguiente:

Las funciones extremas de las válvulas de tres posiciones son idénticas a las de dos posiciones, pero a diferencia de éstas incorporan una posición central adicional. Esta posición podrá ser de centro cerrado, centro abierto o centro a presión. Un centro abierto permite la detención intermedia de un actuador en forma libre, dado que ambas cámaras quedan conectadas a escape en esa posición. Un centro cerrado, por el contrario, permitirá una parada intermedia, pero el cilindro quedara bloqueado por imposibilitarse sus escapes. El centro a presión mantiene alimentadas ambas cámaras, lo que permite detener con precisión un cilindro sin vástago, compensando eventuales pérdidas de aire del circuito. (Microautomacion, 2014)



Válvula con 3 posiciones de
trabajo

Figura 21 Válvula de 3 posiciones.

Fuente: (Microautomacion, 2014)

2.9.4 Cilindro neumático

Los cilindros neumáticos son unidades que transforman la energía potencial del aire comprimido en energía cinética o en fuerzas prensoras. Esto se alcanza por medio del gas comprimido, que es debido a la diferencia de presión. (Gonzales, 2014)

2.9.4.1 Cilindro de simple efecto

El estudio de Bueno, 2014 se encontró lo siguiente:

Se trata de un tubo cilíndrico cerrado dentro del cual hay un émbolo unido a un vástago que se desplaza unido a él. Por un extremo hay un orificio para entrar o salir el aire y en el otro está albergado un muelle que facilita el retorno del vástago. Este tipo de cilindro trabaja en un solo sentido, cuando el aire entra en él. El retroceso y desalojo del aire se produce por la fuerza del muelle que está albergado en el interior del cilindro. (Bueno, 2013)

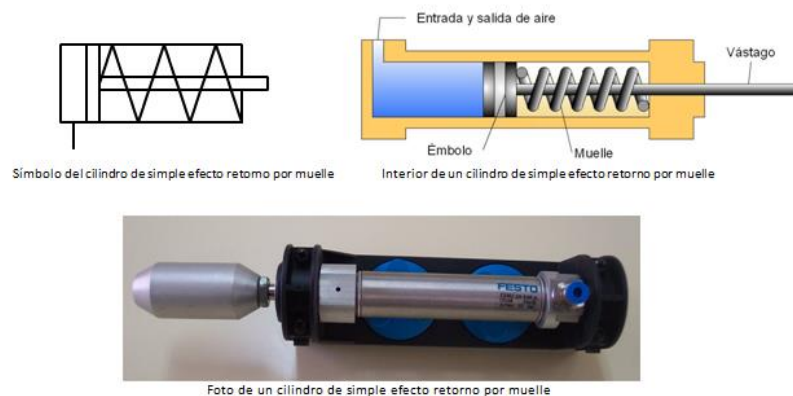


Figura 22 Cilindro simple efecto.

Fuente: (Saramago, 2016)

2.9.4.2 Cilindro de doble efecto

El estudio de Bueno, 2014 se encontró lo siguiente:

Se trata de un tubo cilíndrico cerrado con un diseño muy parecido al cilindro de simple efecto, pero sin el muelle de retorno, el retorno se hace por medio de otra entrada de aire. Este tipo de cilindro trabaja en los dos sentidos, cuando el aire entra en él produce fuerza y desaloja

el aire que está en el otro compartimento. El retroceso y desalojo del aire se produce cuando el aire entra por el otro orificio. (Bueno, 2013)

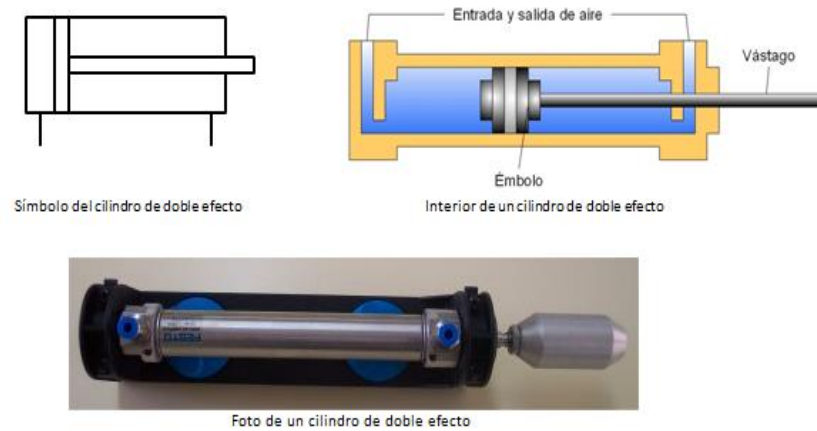


Figura 23 Cilindro doble efecto.

Fuente: (Saramago, 2016)

2.10 UNIDAD DE MANTENIMIENTO.

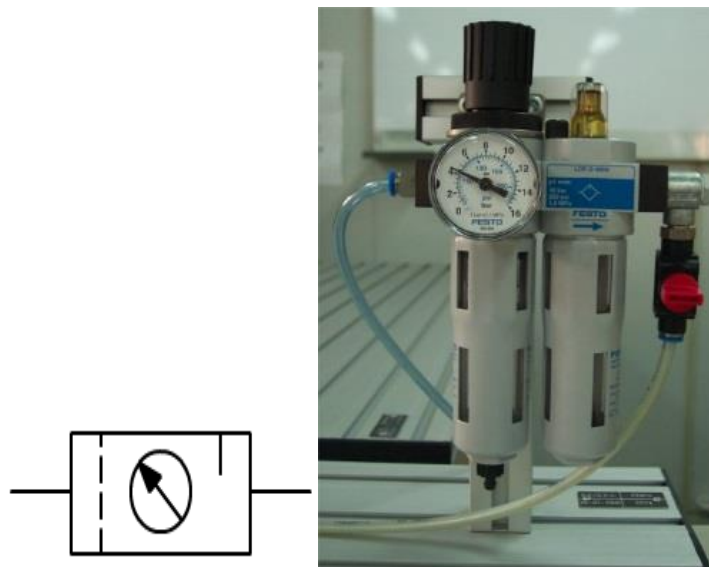


Figura 24 Unidad de mantenimiento.

Fuente: (Bueno, 2013)

Según Tangient, 2016 se encontro lo siguiente:

Los dispositivos conectados en los diferentes puntos de un circuito neumático necesitan recibir aire con una presión uniforme y libre de impurezas. Además, muchos de estos dispositivos tienen elementos móviles que precisan ser lubricados. La preparación del aire comprimido que consumen los dispositivos neumáticos conectados en

diferentes puntos se realiza mediante las llamadas unidades de mantenimiento. Las partículas más grandes, son retenidas por el filtro sinterizado, mientras que los líquidos son desviados al vaso del filtro. El líquido condensado en el vaso o copa del filtro se debe vaciar periódicamente, ya que podría ser arrastrado por la corriente del aire comprimido al circuito. Los filtros más finos, de hasta 0.01 micras, se encargan de filtrar las partículas más pequeñas e incluso mínimas gotas de agua que pudieran quedar en el aire comprimido. Estas unidades están formadas por tres elementos diferentes: el filtro, el regulador y el lubricador.

Filtro.- Tiene como objetivo detener las impurezas que arrastra el aire comprimido (polvo, polen, restos de pequeñas oxidaciones, etc.).

Regulador.- El regulador de presión tiene como misión mantener el aire que utiliza el circuito neumático a una presión constante, independientemente de las variaciones de presión que se produzcan.

Lubricador.- Una vez filtrado, y regulada su presión, el aire comprimido pasa a través del lubricador mezclándose con una fina capa de aceite que arrastra en suspensión hasta las partes móviles de los dispositivos neumáticos. De esta manera son lubricados disminuyendo la fricción y evitando el desgaste. (Tangient, 2016)

CAPÍTULO III

DESARROLLO DEL TEMA.

3.1 Preliminares.

En este capítulo se describirá, de una forma técnica, el desarrollo de la aplicación software que conforma la interfaz gráfica de este prototipo y permite visualizar en tiempo real la situación relativa de los Módulos Remotos unos respecto de otros. De esta manera, la parte de desarrollo del hardware queda completada con un desarrollo software que resulta altamente intuitivo y de fácil manejo.

3.2 Requerimientos mínimos

Los componentes para la implementación de un control inalámbrico para el módulo de electroneumática mediante XBee-Pro fueron los siguientes:

- Dos Módulo XBee pro S1
- Arduino mega
- Arduino uno
- LCD (4x16)
- Módulo de Electroneumática
- Cables UTP flexible
- cuatro pulsadores
- CPU
- Dos cajas de acrílico
- Fuente de 9V DC
- Shield XBee diseñados

Software utilizado:

- Plataforma Arduino
- Proteus 8.1
- XCTU

3.3 Instalación del software X-CTU

XCTU es una aplicación multi-plataforma libre diseñada para permitir a los desarrolladores interactuar con módulos Digi RF a través de una interfaz

gráfica fácil de usar. Incluye nuevas herramientas que hacen más fácil de configurar y probar módulos RF Xbee. (Digi, Digi International Inc, 2016)

Se procede con los siguientes pasos de instalación. Descargar de la página web “<http://www.digi.com/>” (Digi, DIGI, 2016). En la figura 25 de la pantalla DIGI XCTU dar clic en el botón “next”.

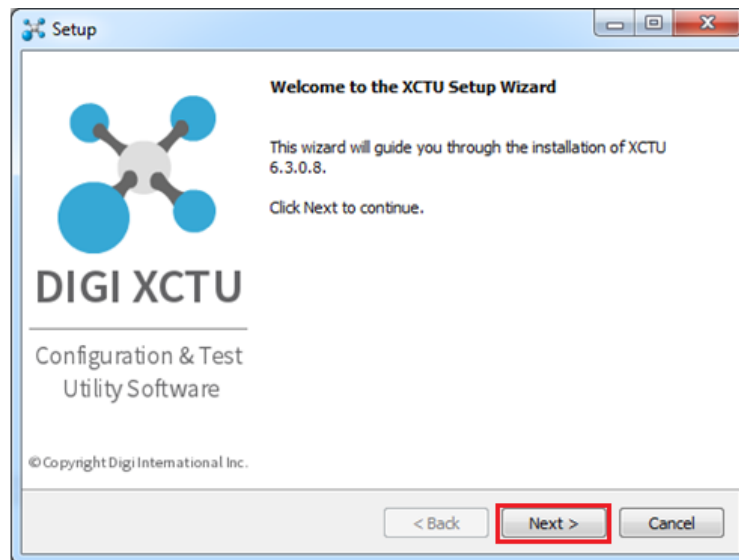


Figura 25 Pantalla DIGI XCTU.

Fuente: (XCTU, 2016)

En la figura 26 de la pantalla License Agreement, señalar “I accept the agreement” y dar clic en el botón “next”.



Figura 26 Activacion de licencia.

Fuente: (XCTU, 2016)

En la pantalla de la figura 27 está configurando el software XCTU al sistema operativo previamente instalado en PC y dar clic en el botón “next”.

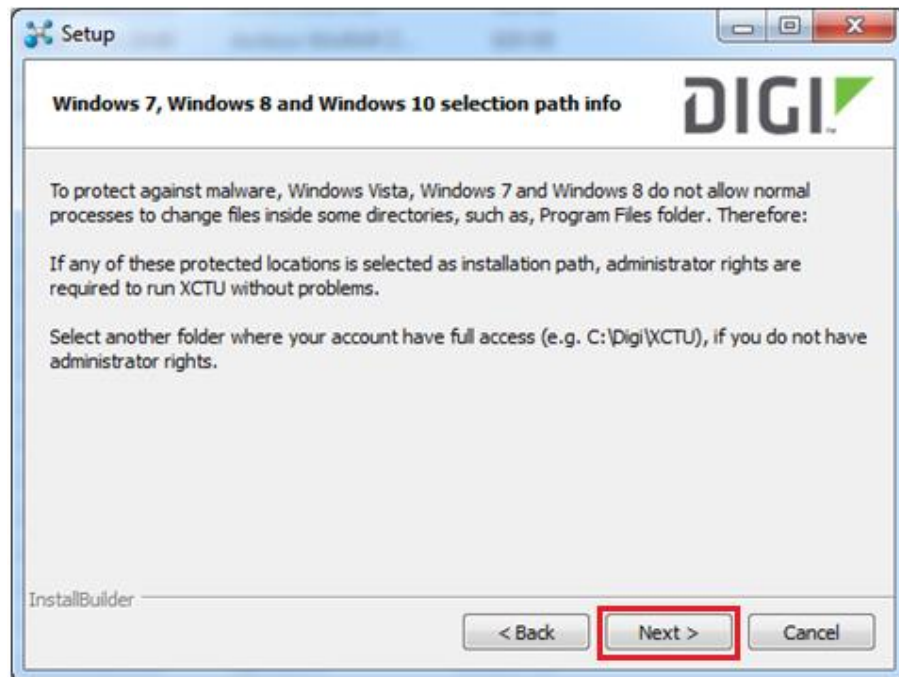


Figura 27 Selección de Sistema Operativo.

Fuente: (XCTU, 2016)

En la pantalla de la figura 28, indica la ruta en donde se instalará el software XCTU.

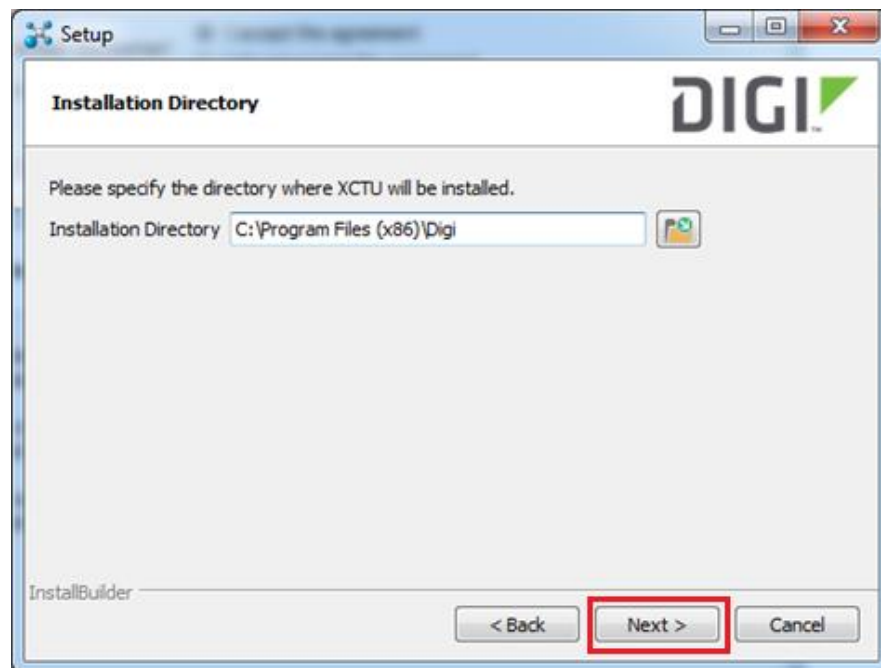


Figura 28 Direccion del software XCTU.

Fuente: (XCTU, 2016)

En la pantalla de la figura 29, indica que está listo para la instalación del software XCTU y dar clic en el botón “next”.



Figura 29 Listo para instalar software.

Fuente: (XCTU, 2016)

En la pantalla de la figura se observa el proceso de la instalación del software XCTU.

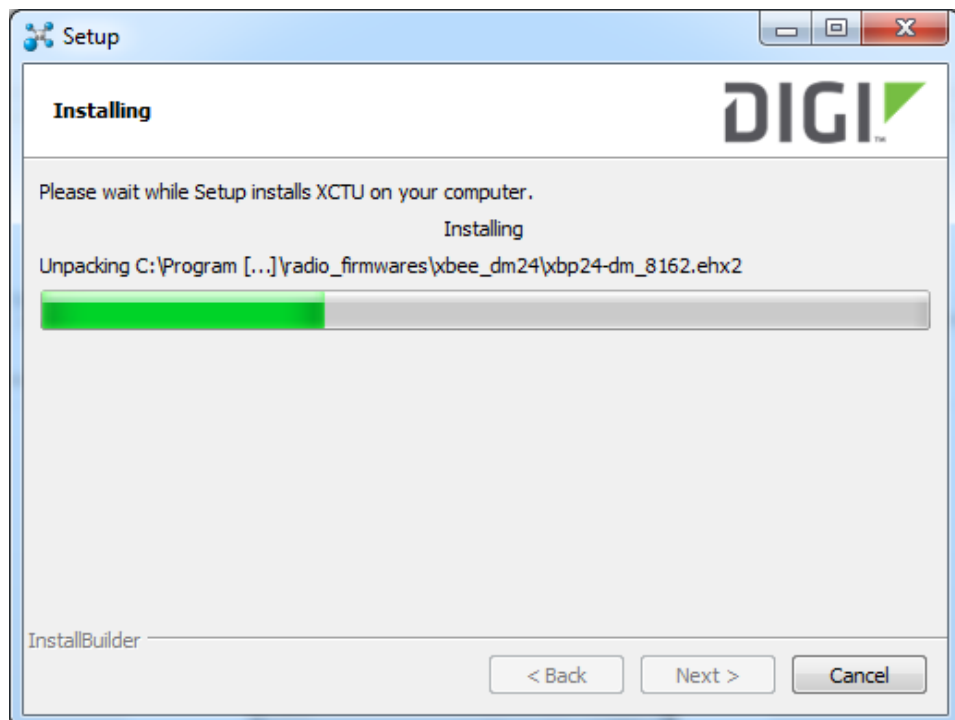


Figura 30 Instalando software.

Fuente: (XCTU, 2016)

En la pantalla de la figura 31, indica que la instalación ha concluido correctamente y dar clic al botón en "Finish".

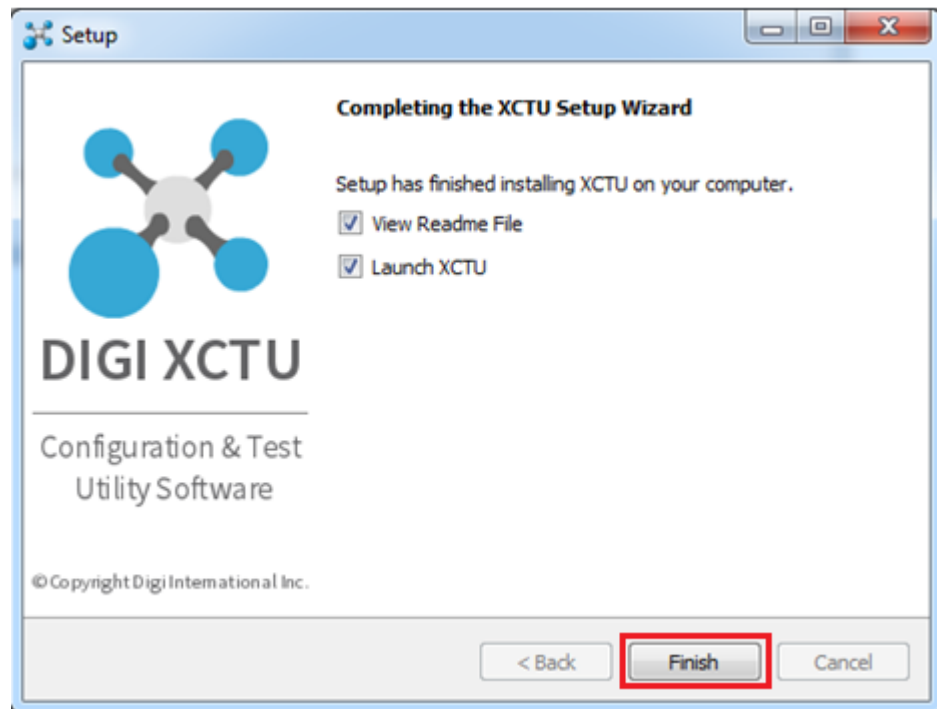


Figura 31 Instalación completada.

Fuente: (XCTU, 2016)

3.4 Configuración del módulo XBee pro mediante el software x-ctu

Para la configuración de los módulos XBee Pro S1 se utilizó el software X-CTU (Figura 32), proveído por la compañía Digi y se procede con los siguientes pasos:

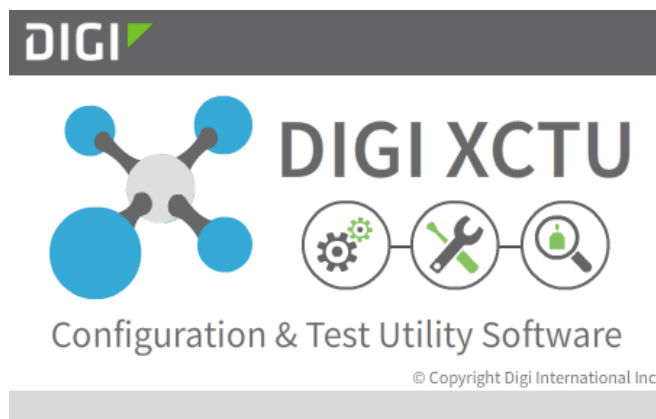


Figura 32 Logo del software XCTU.

Fuente: (XCTU, 2016)

En la pantalla de la figura 33, se observa en la parte izquierda el panel para añadir los módulos Xbee y en la parte derecha la consola para la configuración del Xbee.

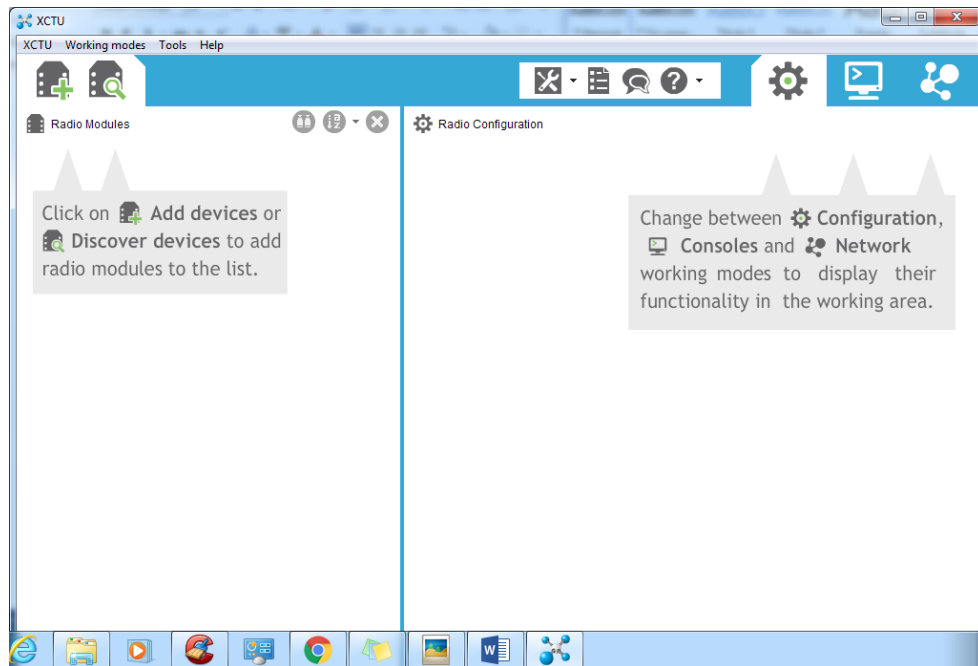


Figura 33 Pantalla del software XCTU.

Fuente: (XCTU, 2016)

3.4.1 Secciones principales del software XCTU

3.4.1.1 Barra de menús

La barra de menú se encuentra en la parte superior de la aplicación. Se utiliza para acceder a todas las características XCTU, herramientas y modos de trabajo. (Digi, 2016)

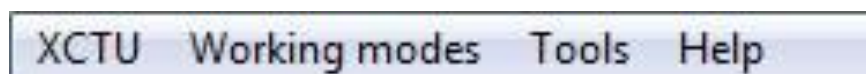


Figura 34 Barra de menús.

Fuente: (XCTU, 2016)

3.4.1.2 Barra de herramientas principal

La barra de herramientas principal se encuentra en la parte superior de la aplicación y se divide en tres secciones. (Digi, 2016)



Figura 35 Barra principal de herramientas.

Fuente: (XCTU, 2016)

La primera sección contiene dos iconos que se utilizan para agregar módulos de radio a la lista de módulos de radio. (Digi, 2016)



Figura 36 Iconos para agragar modulos.

Fuente: (XCTU, 2016)

La segunda sección contiene la funcionalidad XCTU estática que no requiere un módulo de radio. Esta sección incluye las herramientas XCTU, la configuración XCTU, el formulario de comentarios y la ayuda y funciones actualizaciones. (Digi, 2016)



Figura 37 Funcionalidad XCTU.

Fuente: (XCTU, 2016)

La tercera sección contiene las fichas correspondientes a los tres modos de trabajo XCTU. Para utilizar esta funcionalidad, debe haber agregado uno o más módulos de radio a la lista. (Digi, 2016)



Figura 38 Modos de trabajo XCTU.

Fuente: (XCTU, 2016)

3.4.1.3 Lista de dispositivos

El estudio de Digi, 2016 se encontró lo siguiente:

La lista de módulos de radio, o la lista de dispositivos, están situados en el lado izquierdo de la herramienta y muestra los módulos de radio que están conectados a su ordenador. Si conoce la configuración del puerto serie de un módulo de radio, puede agregarlo a la lista directamente. También puede utilizar la función de descubrimiento de XCTU para encontrar módulos de radio conectados a su PC y añadirlos a la lista.

Dependiendo del protocolo de los módulos de radio locales se sumaron, también se pueden añadir módulos de radio remotos a la lista utilizando la función de búsqueda del módulo. (Digi, 2016)

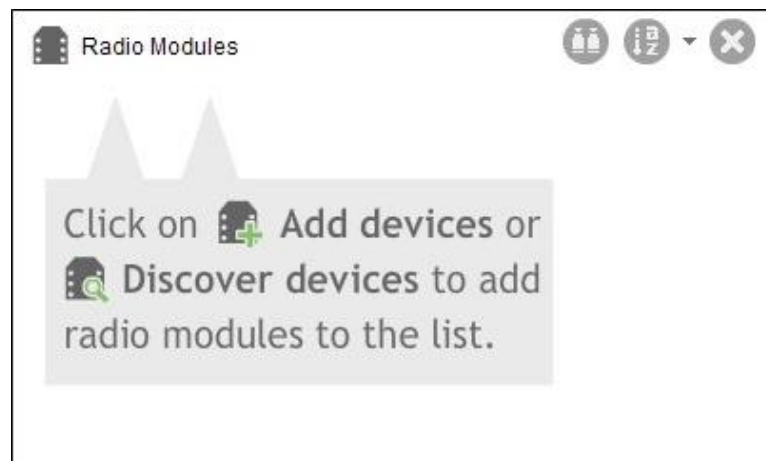


Figura 39 Módulos de radio locales.

Fuente: (XCTU, 2016)

3.4.1.4 Área de trabajo

El estudio de Digi, 2016 se encontró lo siguiente:

El área de trabajo es la sección más grande y se encuentra en el lado derecho de la aplicación. El contenido de la zona de trabajo depende del modo de trabajo seleccionado en la barra de herramientas. Para interactuar con los controles que aparecen en el área de trabajo, debe haber agregado uno o más módulos de radio a la lista y uno de los módulos debe estar seleccionado. (Digi, 2016)

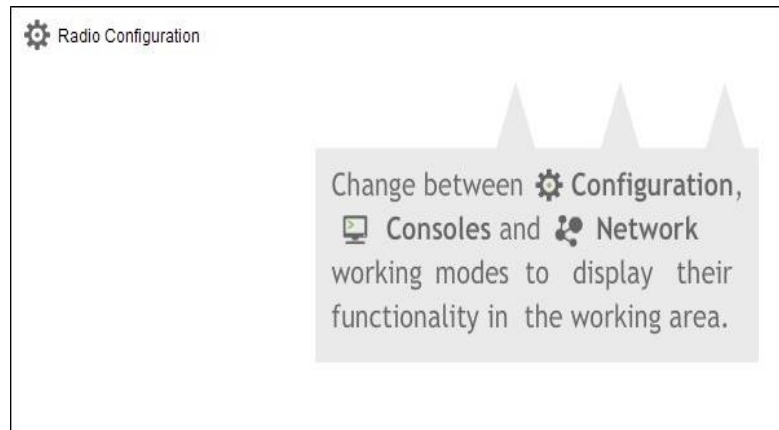


Figura 40 Area de trabajo.

Fuente: (XCTU, 2016)

3.4.2 INSTALAR DIGI_USE_RF_Drivers

El estudio de Digi, DIGI_support, 2016 se encontró lo siguiente:

Al instalar DIGI_USE_RF_Drivers, permitirá configurar y operar el módulo XBee PRO desde el computador, para lo cual se coloca dicho modulo en el programador XBee Pro y se conecta mediante el puerto USB al computador, para algunas versiones de Windows los drivers serán instalados de manera automática, mientras que para otros se debe realizar el proceso de instalación de manera manual desde el administrador de dispositivos. (Digi, DIGI_support, 2016)

3.4.3 CONFIGURAR LOS MÓDULOS XBEE PRO S1

Una vez realizado el proceso anterior, en el software X-CTU y dar click en añadir dispositivos y se observa el puerto que se acaba de instalar en este caso el COM 3. Para asegurar que todo está operativo, se presiona el botón "Finish" (Figura 41) y se desplegará una pequeña ventana indicando que el módulo XBee se está conectando con la PC.

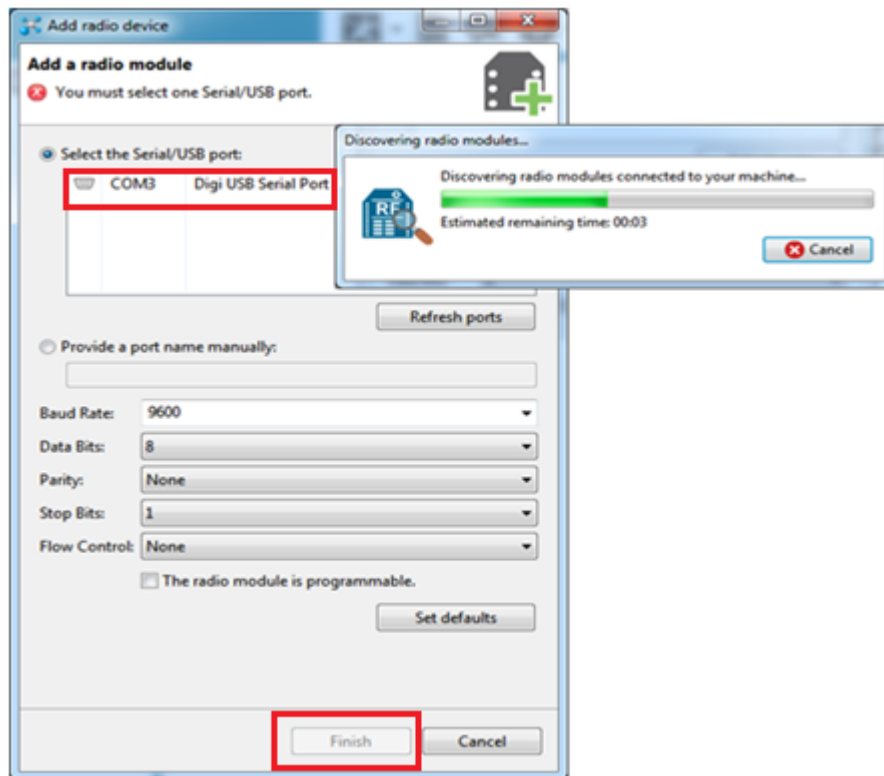


Figura 41 Seleccionar puerto serial.

Fuente: (XCTU, 2016)

En la figura 42, para programar dar clic en ventana de color naranja y se desplegara otra ventana indicando que está leyendo los parámetros que tienes el módulo Xbee maestro con MAC: 0013A200406E285F.

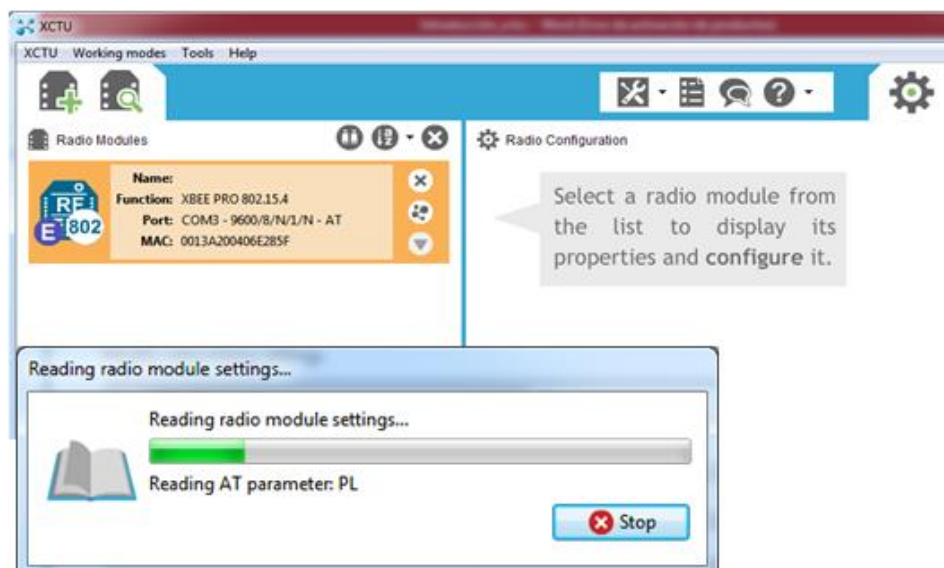


Figura 42 Paramatros Xbee maestro con MAC: 0013A200406E285F.

Fuente: (XCTU, 2016)

Como se muestra en la figura 43 se procede de igual manera con el otro módulo XBee esclavo con MAC 0013A200406AEF16.

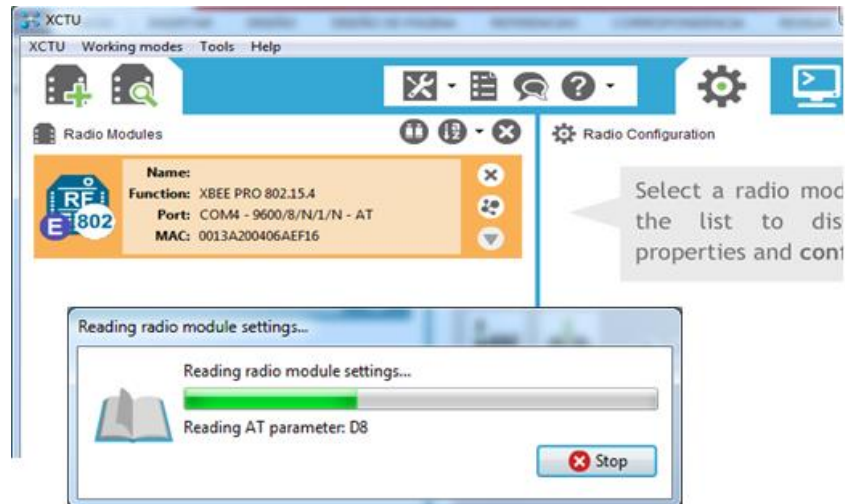


Figura 43 Parametros Xbee esclavo con MAC 0013A200406AEF16.

Fuente: (XCTU, 2016)

En la figura 44 indica los parámetros a modifica del módulo Xbee maestro en este caso ID, DH, DL, MY, PL y para guardar los cambios damos clic en “write”.

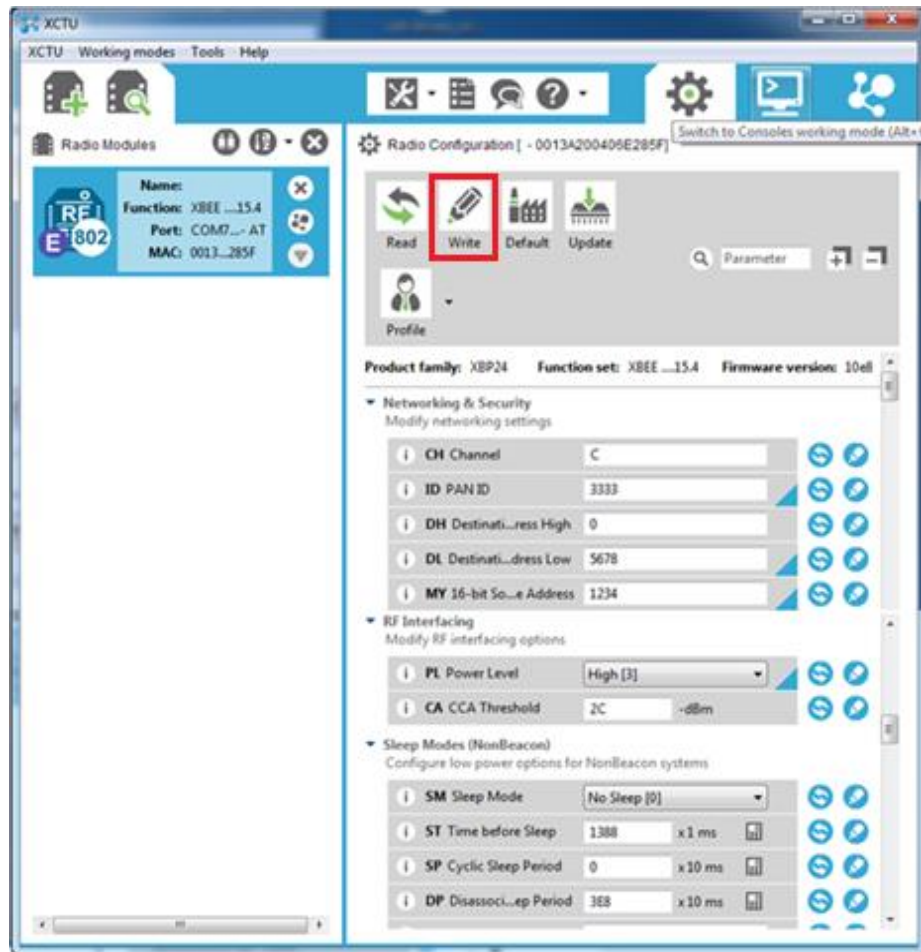


Figura 44 Parámetros modificados Xbee maestro con MAC 0013A200406E285F

Fuente: (XCTU, 2016)

Tabla 5

Parámetros del Xbee maestro y Xbee esclavo.

	Xbee Maestro	Xbee Esclavo
CH	C	C
ID	3333	3333
DH	0	0
DL	5678	1234
MY	1234	5678
PL	3	3

Tabla 6

Descripción de comandos.

Comando	Nombre y descripción
CH	Canal. Set/leer el número del canal utilizado para transmitir y recibir datos entre módulos RF (usa Protocolo 802.15.4 números de canal).
ID	PAN ID. Set/leer el PAN (red de área personal) ID. Utilice 0xFFFF para difundir mensajes a todos los recipientes.
DH	Dirección de destino de Alta. Set/leer los 32 bits superiores de la dirección de destino de 64 bits. Cuando se combina con DL, define la dirección de destino utilizada para la transmisión. Para transmitir mediante una dirección de 16 bits, ajuste a cero el parámetro DH y DL menor que 0xFFFF. 0x000000000000FFFF es la dirección de broadcast para el PAN.
DL	Bajo la dirección de destino. Set/leer los 32 bits inferiores de la dirección de destino de 64 bits. Cuando se combina con DH, DL define el destino dirección utilizada para la transmisión. Para transmitir mediante una dirección de 16 bits, ajuste a cero el parámetro DH y DL menor que 0xFFFF.
MY	Dirección de origen de 16 bits. Set/leer el módulo RF de 16 bits de la dirección de origen. Configurar mi = 0xFFFF para deshabilitar la recepción de paquetes con direcciones de 16 bits. Dirección de origen de 64 bits (número de serie) y la dirección de broadcast (0x000000000000FFFF) siempre está activado.
PL	Nivel de energía. Seleccionar/leer el nivel de potencia al cual el módulo RF transmite potencia conducida. 0 - 4 (XBee / XBee-PRO); 0 = 10/10 dBm; 1 = 6/12 dBm 2 = 4/14 dBm; 3 = 2/16 dBm; 4 = 0/18 dBm

Fuente: (Digi, 2016)

En la figura 45, se indica los parámetros modificados del módulo XBee esclavo para la comunicación inalámbrica.

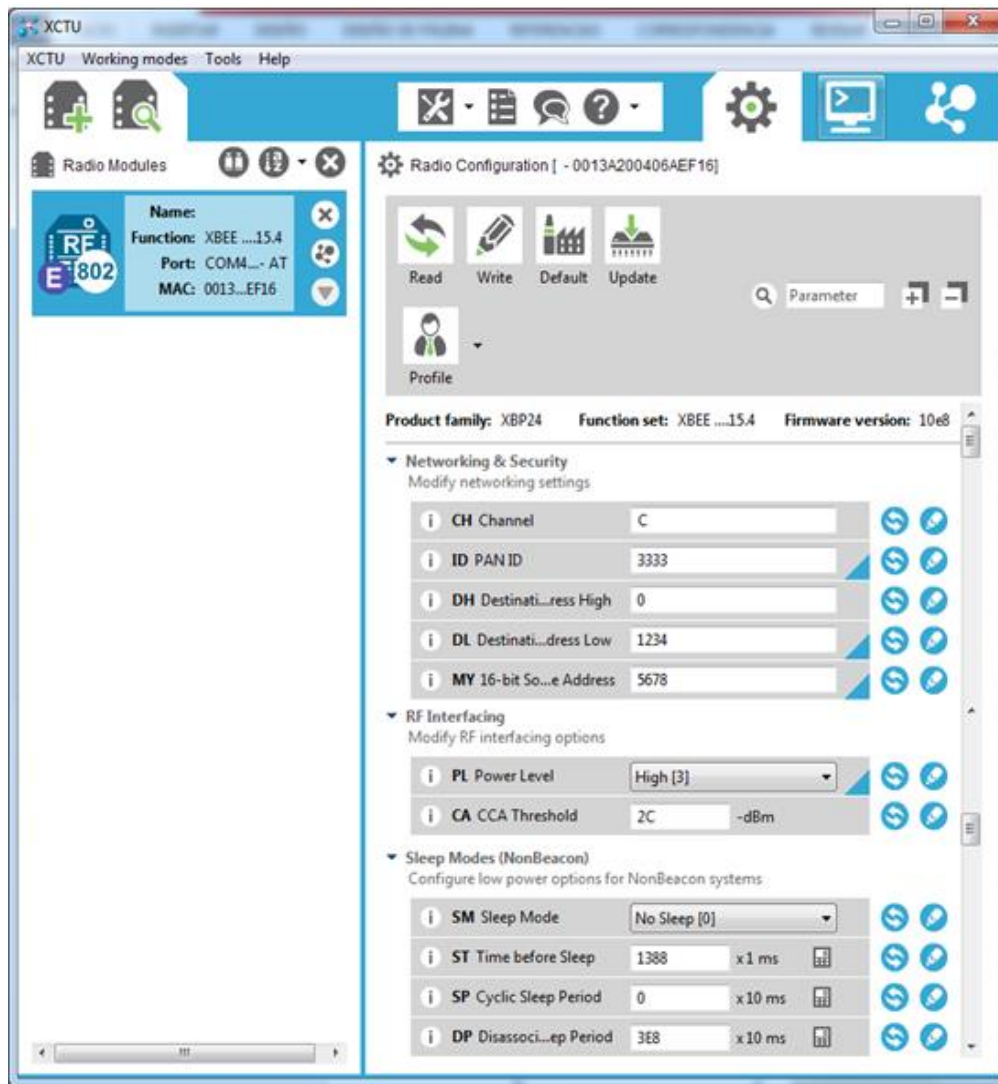


Figura 45 Parametros modificados Xbee Pro esclavo con MAC: 0013A200406AEF16.

Fuente: (XCTU, 2016)

Para verificar si hay comunicación entre los dos módulos Xbee, se realizó de la siguiente forma. Dar clic en la pantalla que se indica del cuadro rojo, lo cual abre otra pantalla en la parte izquierda la que se llama “Consola” como se indica en la figura 46.

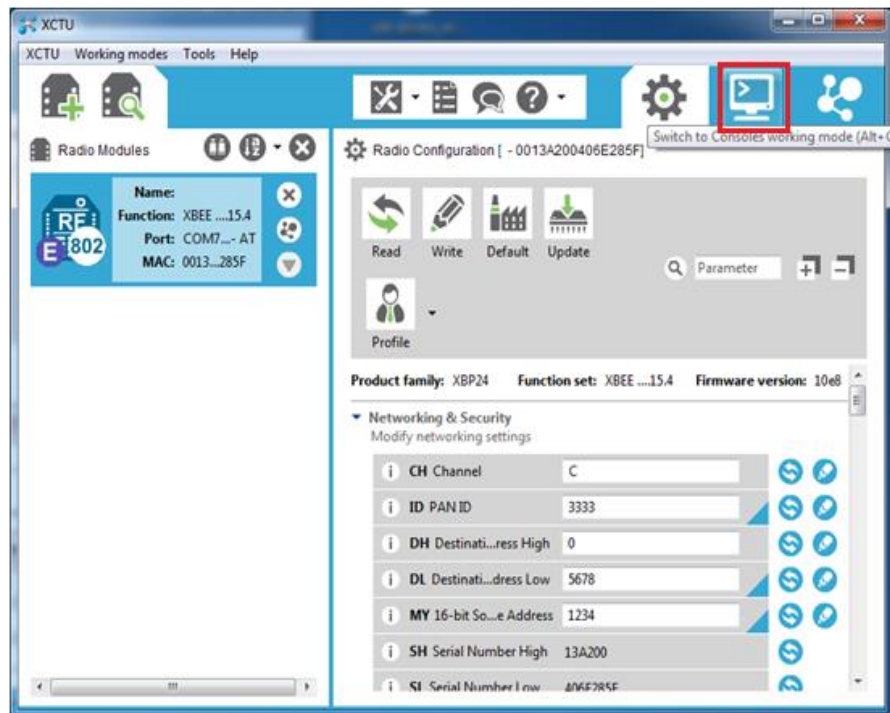


Figura 46 Cambio a modo consola del XCTU.

Fuente: (XCTU, 2016)

En la figura 47 se observa la pantalla de consola y para conectar la comunicación serial dar clic en el icono del enchufe.

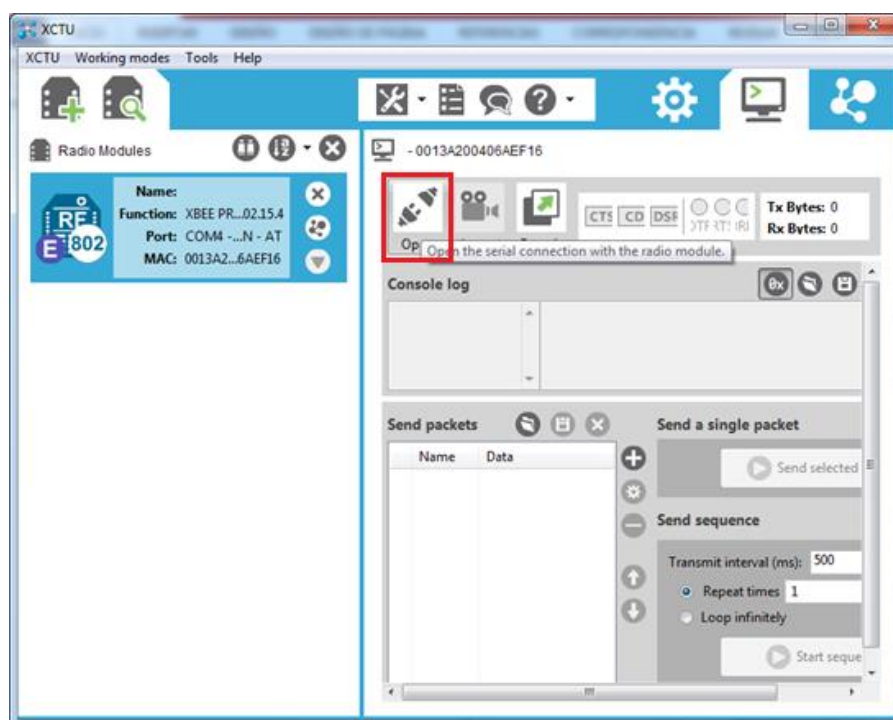


Figura 47 Consola de XCTU.

Fuente: (XCTU, 2016)

En la figura 48 se observa que el módulo Xbee Pro maestro con MAC: 0013A200406E285F está listo para el control inalámbrico y en la figura 49 se observa el módulo Xbee Pro esclavo con MAC 0013A200406AEF16 listo.

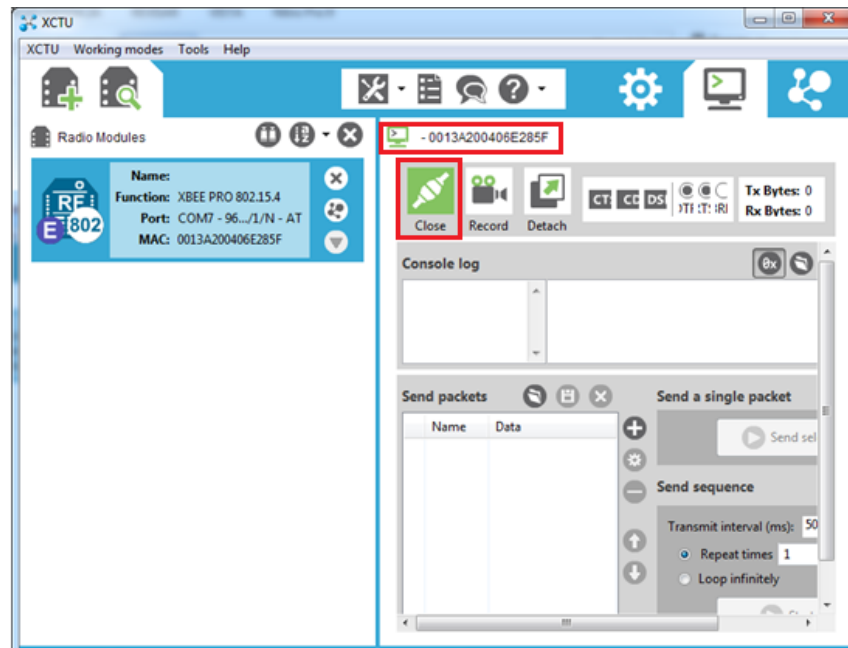


Figura 48 Control inalámbrico Xbee Pro maestro con MAC 0013A200406E285F.

Fuente: (XCTU, 2016)

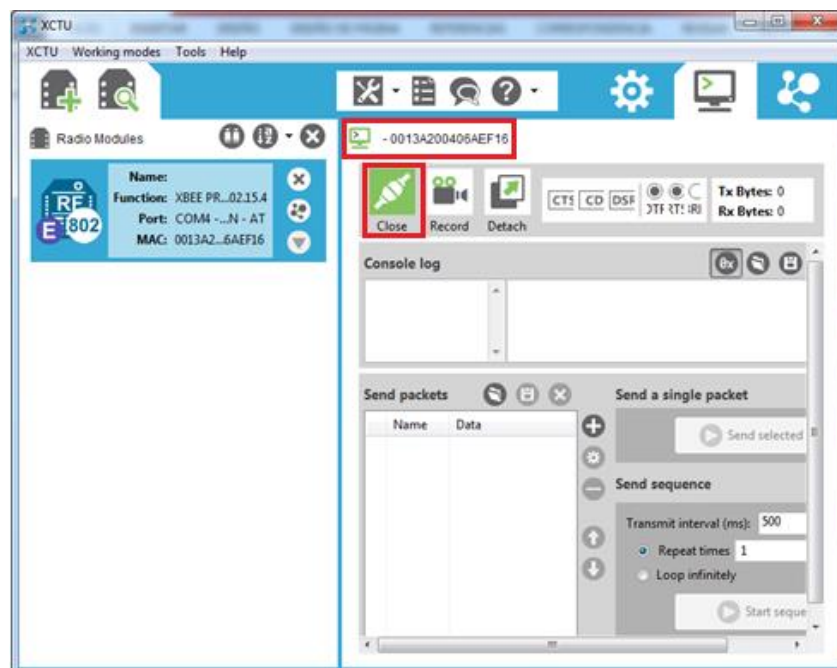


Figura 49 Control inalámbrico Xbee Pro esclavo con MAC: 0013A200406AEF16.

Fuente: (XCTU, 2016)

En la figura 50, se observa la consola del XBee maestro y escribir la palabra “hola” para transmitir.

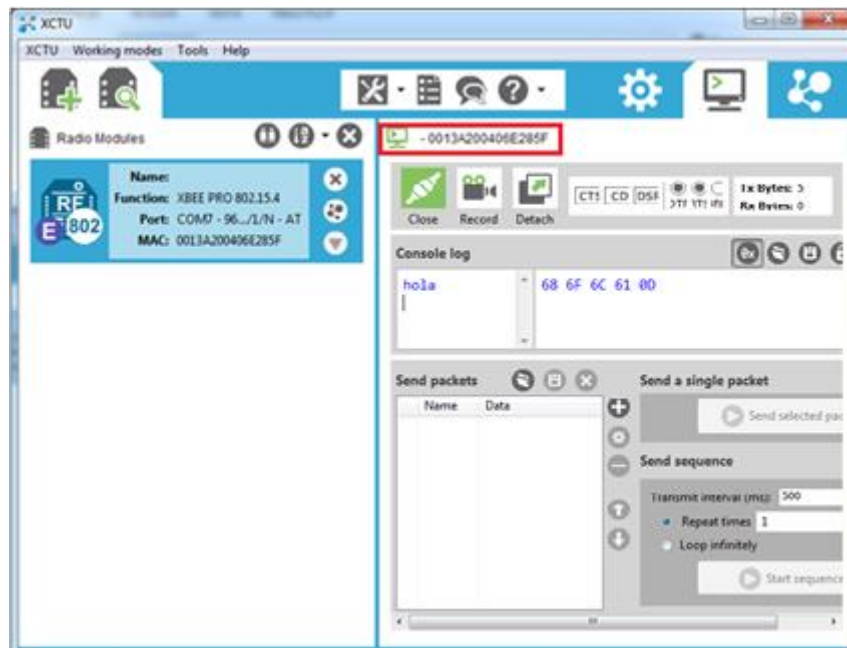


Figura 50 Tx de datos del Xbee Maestro con MAC 0013A200406E285F.

Fuente: (XCTU, 2016)

En la figura 51, se observa que la recepción de la palabra “hola” en el módulo XBee esclavo.

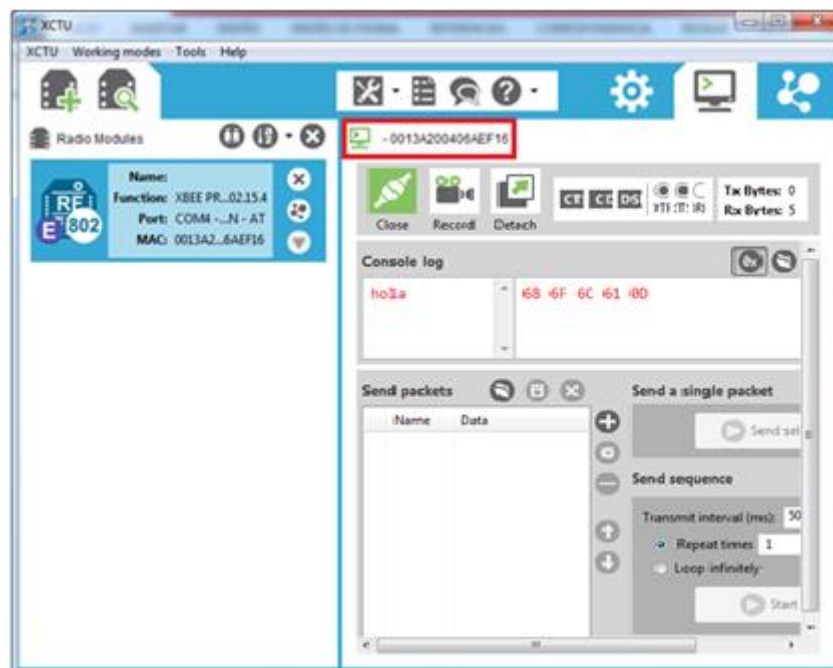


Figura 51 Rx de datos del Xbee esclavo con MAC: 0013A200406AEF16.

Fuente: (XCTU, 2016)

Como es una comunicación serial dúplex, el xbee esclavo puede enviar datos como se muestra en la figura 52 y figura 53.

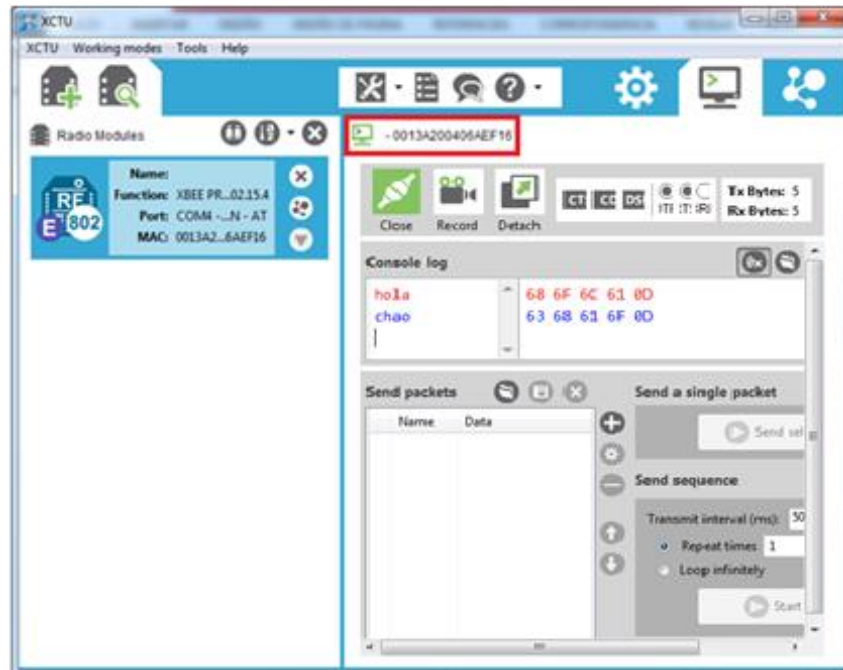


Figura 52 Tx de datos del Xbee esclavo con MAC: 0013A200406AEF16.

Fuente: (XCTU, 2016)

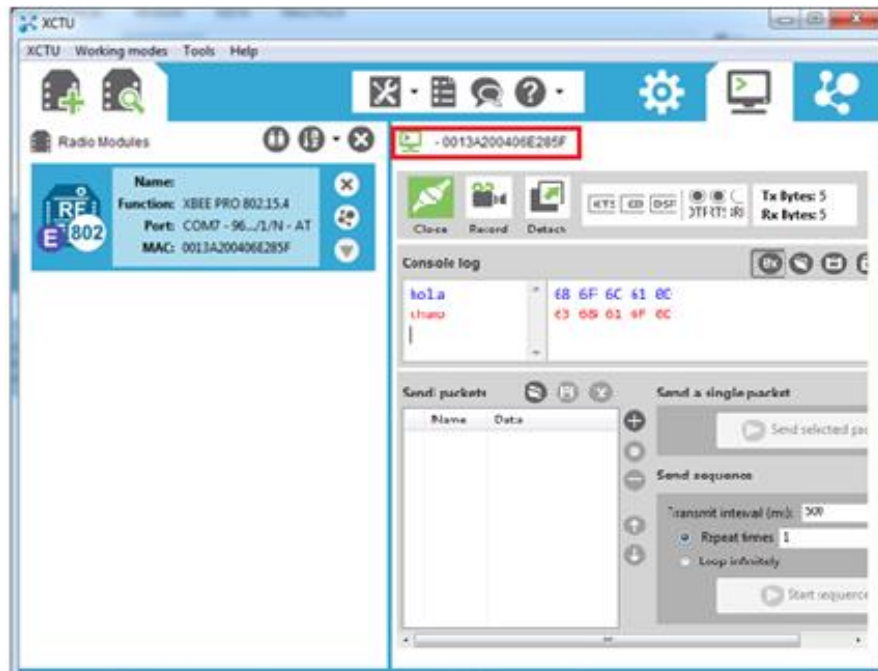


Figura 53 Rx de datos del Xbee Maestro con MAC 0013A200406E285F.

Fuente: (XCTU, 2016)

3.5 Configuración Arduino mega

Para conectar y programar la tarjeta Arduino se siguen los siguientes pasos:

1) Descargar la plataforma arduino

Obtener la última versión del Arduino, descomprimir el archivo descargado en el disco "C" (Arduino, Arduino.cc, 2016)

2) Conexión de la tarjeta

Al conectar el Arduino Mega 2560, automáticamente se alimenta de la conexión USB al ordenador como se indica en la figura 54, si no utilice una fuente de alimentación externa, al realizar esto el led de alimentación de color verde se encenderá.



Figura 54 Conexión USB entre la tarjeta Arduino Mega y la PC.

Fuente: (Alvarez, 2014)

3) Instalación de los controladores

Desde el administrador de dispositivos busque el software del controlador en el equipo e instálelo.

4) Inicie y crear la aplicación arduino

Se procede a realizar la programación en el Arduino, en el cual se procesarán los datos de entrada (pulsadores), los datos a visualizar (LCD) y los datos de salida (Módulo xbee). Creado el programa dar clic en "salvar" y para saber si está bien el código dar clic en "verificar" como se indica en la figura 55.

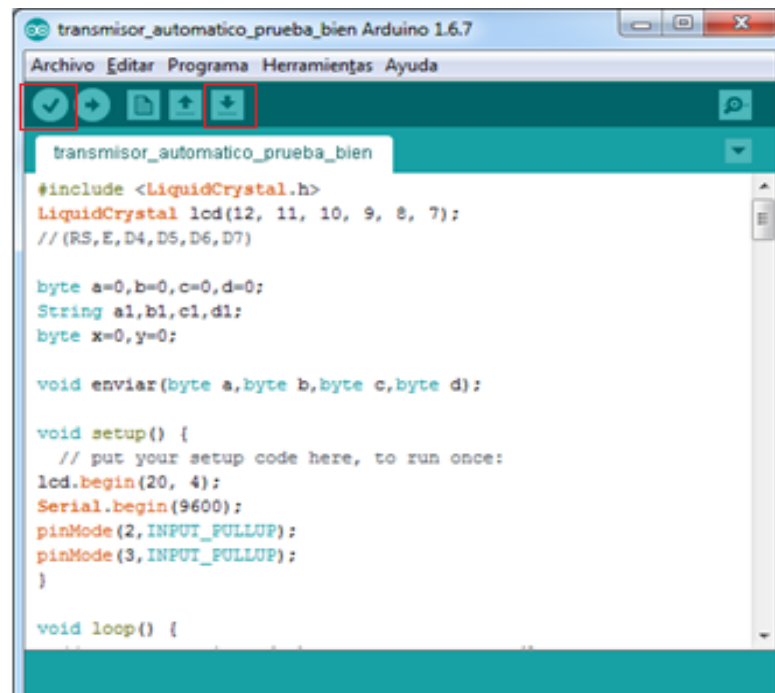


Figura 55 Código transmisor en la plataforma Arduino.

Fuente: (Arduino, 2016)

5) Selección de la tarjeta arduino mega

Se dirige a la pestaña de Herramientas, se selecciona Tarjeta y después Arduino Mega 2560, como se indica en la figura 56

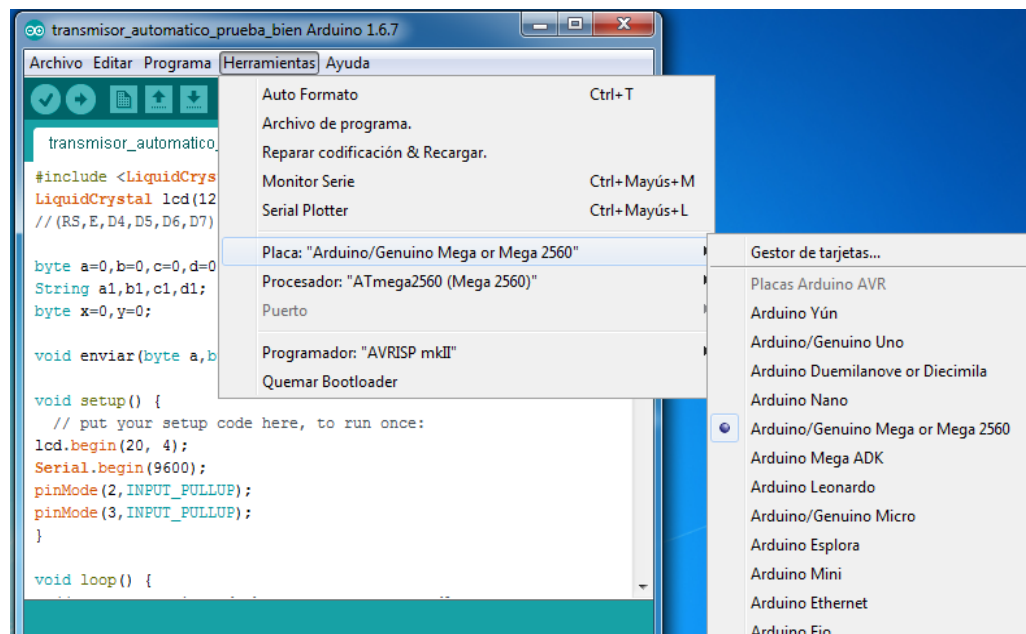


Figura 56 Selección placa arduino mega.

Fuente: (Arduino, 2016)

6) Selección del puerto serial

Desde el menú Herramientas seleccione el puerto serial con el que está trabajando en este caso el “COM 6” como se muestra en la figura 57.

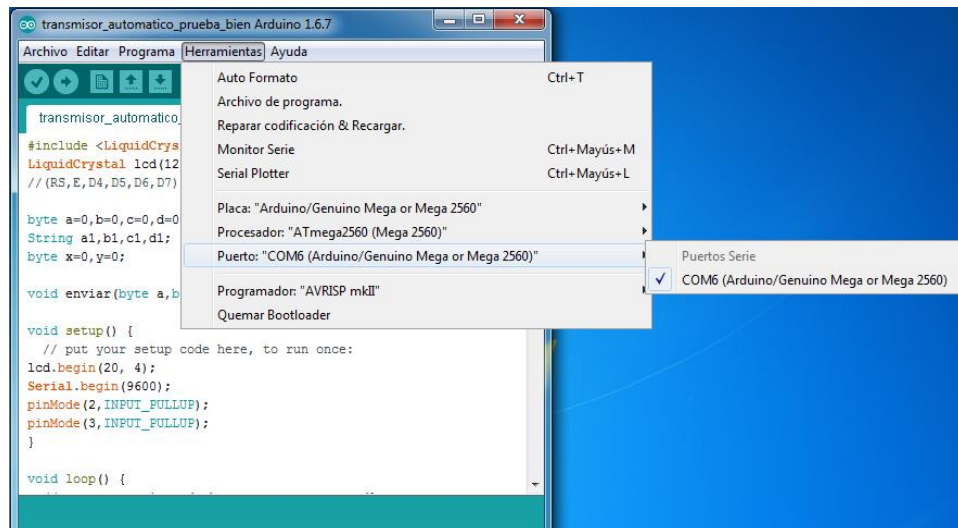


Figura 57 Puerto serial arduino mega.

Fuente: (Arduino, 2016)

7) Cargar el programa

Simplemente haga clic en el botón “Cargar”, espere unos segundos, si la carga se realiza correctamente aparecerá el mensaje de “Carga Terminada” en la barra de estado, como se observa en la figura 58.

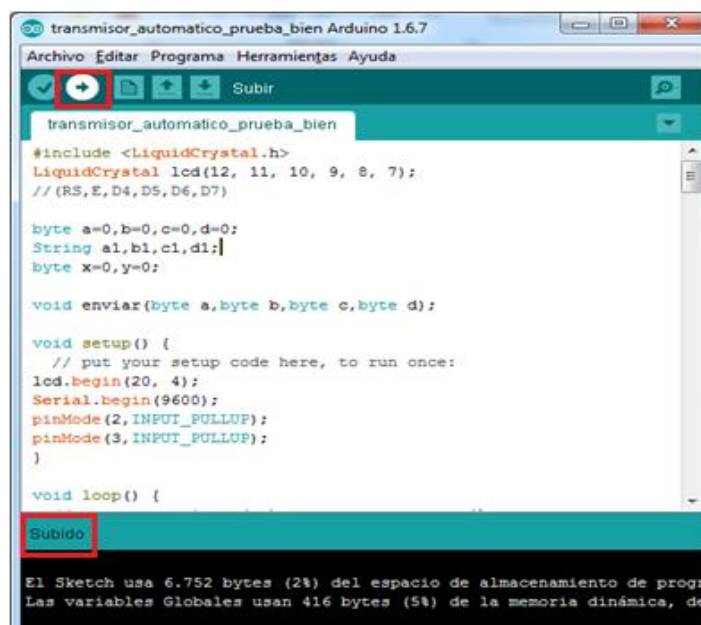


Figura 58 Cargar el programa a la placa arduino mega.

Fuente: (Arduino, 2016)

3.6 Configuración Arduino uno

Para conectar y programar la tarjeta Arduino se siguen los siguientes pasos:

1) Descargar la plataforma arduino

Obtener la última versión del Arduino, descomprimir el archivo descargado en el disco "C" (Arduino, Arduino.cc, 2016)

2) Conexión de la tarjeta

Al conectar el Arduino UNO, automáticamente se alimenta de la conexión USB al ordenador como se indica en la figura 59, si no utilice una fuente de alimentación externa, al realizar esto el led de alimentación de color verde se encenderá.



Figura 59 Conexión USB entre la tarjeta Arduino Uno y la PC.

Fuente: (Alvarez, 2014)

3) Instalación de los controladores

Desde el administrador de dispositivos busque el software del controlador en el equipo e instálelo.

4) Inicie y crear la aplicación arduino

Se procede a realizar la programación en el Arduino, en el cual se procesarán los datos de entrada (datos enviados por módulo xbee maestro) y los datos de salida (control de electroválvulas). Creado el programa dar clic en el botón "salvar" y para saber si está bien el código dar clic en el botón "verificar" como se indica en la figura 60.

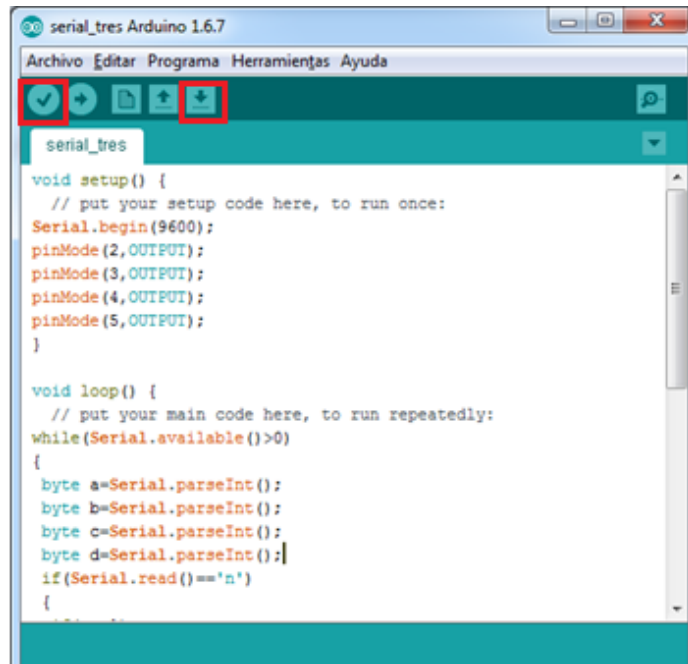


Figura 60 Código del RX en la plataforma arduino.

Fuente: (Arduino, 2016)

5) Selección de la tarjeta arduino uno

Se dirige a la pestaña de Herramientas, se selecciona Tarjeta y después Arduino Uno, como se indica en la figura 61

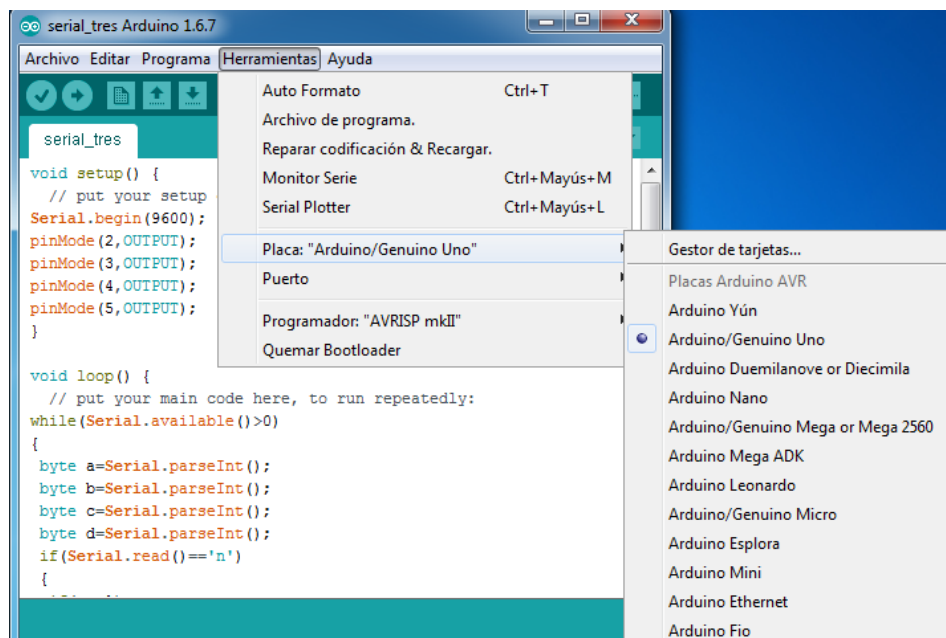


Figura 61 Selección de la tarjeta arduino uno.

Fuente: (Arduino, 2016)

6) Selección del puerto serial

Desde el menú Herramientas seleccione el puerto serial con el que está trabajando en este caso el “COM 6” como se muestra en la figura 62.

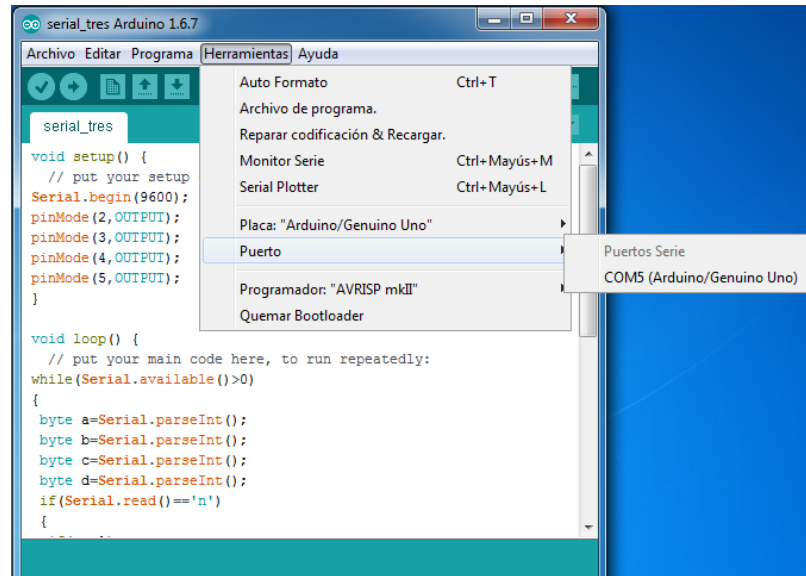


Figura 62 Puerto serial del arduino uno.

Fuente: (Arduino, 2016)

7) Cargar el programa

Simplemente haga clic en el botón “Cargar”, espere unos segundos, si la carga se realiza correctamente aparecerá el mensaje de “subiendo” en la barra de estado, como se observa en la figura 63.

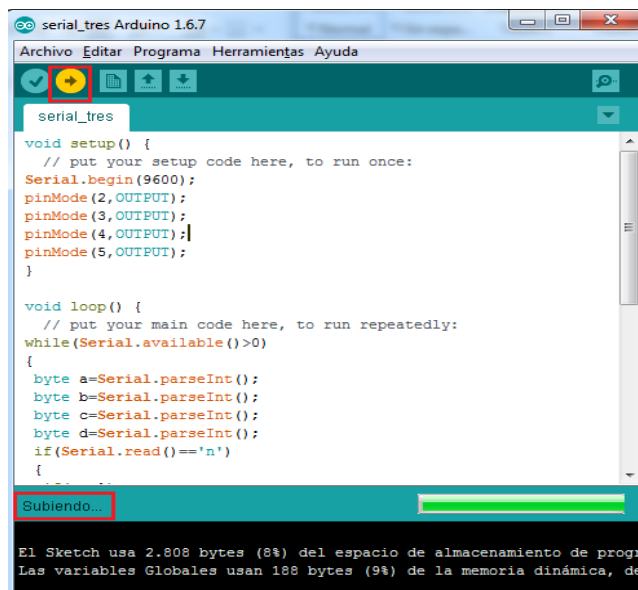


Figura 63 Subiendo el Programa a la placa arduino uno.

Fuente: (Arduino, 2016)

3.7 Código de programación del control Inalámbrico

3.7.1 Aplicación manual

Al presionar el pulsador conectado en el pin D2 (Válvula Principal) del arduino mega y se muestra en la pantalla LCD la palabra “válvula encendida” cuando la entrada está en 1 lógica activándose la electroválvula y la palabra “válvula apagada” cuando la entrada está en 0 lógica desactivándose la electroválvula, al presionar el pulsador conectado en el pin D3 (Cilindro 1) se observa en la LCD la palabra “Cilindro 1 Afuera” cuando la entrada está en 1 lógica y la palabra “Cilindro 1 Adentro” cuando la entrada está en 0 lógica, al presionar el pulsador conectado en el pin D4 (Cilindro 2) se observa en la LCD la palabra “Cilindro 2 Afuera” cuando la entrada está en 1 lógica y la palabra “Cilindro 2 Adentro” cuando la entrada está en 0 lógica, al presionar el pulsador conectado en el pin D5 (Cilindro 3) se observa en la LCD la palabra “Cilindro 3 Afuera” cuando la entrada está en 1 lógica y la palabra “Cilindro 3 Adentro” cuando la entrada está en 0 lógica.

3.7.1.1 Xbee Maestro Arduino Mega

```
// librería para la lcd 4x20
#include <LiquidCrystal.h>
// descripción de los pines de comunicación a utilizar
//(RS,E,D4,D5,D6,D7)
LiquidCrystal lcd(12, 11, 10, 9, 8, 7);
// Declaracion de variables
byte r=0,s=0,t=0,u=0;
byte a,b,c,d;
String a1,b1,c1,d1;

void setup() {
// Inicializa la LCD
lcd.begin(20, 4);
// Inicia la comunicación serial a una la velocidad de datos a 9600 bps
Serial.begin(9600);
// Configura el pin 2 como entrada y habilita la resistencia interna de pull-up
pinMode(2,INPUT_PULLUP);
```

```

// Configura el pin 3 como entrada y habilita la resistencia interna de pull-up
pinMode(3,INPUT_PULLUP);
// Configura el pin 4 como entrada y habilita la resistencia interna de pull-up
pinMode(4,INPUT_PULLUP);
// Configura el pin 5 como entrada y habilita la resistencia interna de pull-up
pinMode(5,INPUT_PULLUP);
}

void loop() {
// Lee el valor del pin digital 2 especificado, ya sea alta o baja.
// Si r<1 y a=1 entonces se mostrara el mensaje válvula encendida en la LCD
// caso contrario si r=0 y a=0 se mostrara válvula apagada
if (!digitalRead(2))
{
  if (r<1)
  {
    r++;
    a=1;
    a1="Encendida";
  }
  else
  {
    r=0;
    a=0;
    a1="Apagada ";
  }
  // retardo de 3s
  delay(300);
}

// Lee el valor del pin digital 3 especificado, ya sea alta o baja.
// Si s<1 y b=1 entonces se mostrara el mensaje cilindro afuera en la LCD
// caso contrario si s=0 y b=0 se mostrara cilindro adentro
if (!digitalRead(3))
{

```

```
if (s<1)
{
s++;
b=1;
b1="Afuera ";
}
else
{
s=0;
b=0;
b1="Adentro";
}
// 3 segundos de retardo
delay(300); }
// Lee el valor del pin digital 4 especificado, ya sea alta o baja.
// Si t<1 y c=1 entonces se mostrara el mensaje cilindro afuera en la LCD caso
contrario si t=0 y c=0 se mostrara cilindro adentro
if (!digitalRead(4))
{
if (t<1)
{
t++;
c=1;
c1="Afuera ";
}
else
{
t=0;
c=0;
c1="Adentro";
}
// 3 segundos de retardo
delay(300);
}
```

```

// Lee el valor del pin digital 5 especificado, ya sea alta o baja.
// Si u<1 y d=1 entonces se mostrara el mensaje cilindro afuera en la LCD
// caso contrario si u=0 y d=0 se mostrara cilindro adentro
if (!digitalRead(5))
{
  if (u<1)
  {
    u++;
    d=1;
    d1="Afuera ";
  }
  else
  {
    u=0;
    d=0;
    d1="Adentro";
  }
  // 3 segundos de retardo
  delay(300);
}

```

3.7.1.2 Xbee Esclavo Arduino Uno:

```

void setup() {
// Inicia la comunicación serial a una la velocidad de datos a 9600 bps
Serial.begin(9600);
//Declaracion del pin 2 como salida, activarlo aplicando 5v en el pin y
//desactivarlo con 0V
pinMode(2,OUTPUT);
//Declaracion del pin 3 como salida, activarlo aplicando 5v en el pin y
//desactivarlo con 0V
pinMode(3,OUTPUT);
//Declaracion del pin 4 como salida, activarlo aplicando 5v en el pin y
//desactivarlo con 0V
pinMode(4,OUTPUT);
}

```



```
//Declaracion del pin 5 como salida, activarlo aplicando 5v en el pin y
desactivarlo con 0V
pinMode(5,OUTPUT);
}
void loop() {
// Enviar datos sólo cuando se reciben datos
while(Serial.available(>0)
{
// Devuelve el primer número entero válido (largo) de la memoria intermedia
serie de la variable tipo "byte".
//Los caracteres que no sean números enteros (o el signo menos) se omiten
byte a=Serial.parseInt();
byte d=Serial.parseInt();
// Lee datos serie entrantes
if(Serial.read()=='n')
{
// Si a=1 el pin 2 se pone en alto y se enciende el Led
if(a==1)
digitalWrite(2,HIGH);
// Si a=0 el pin 2 se pone en bajo y se apaga el Led
if(a==0)
digitalWrite(2,LOW);
// Si b=1 el pin 3 se pone en alto y se enciende el Led
if(b==1)
digitalWrite(3,HIGH);
// Si b=0 el pin 3 se pone en bajo y se apaga el Led
if(b==0)
digitalWrite(3,LOW);
// Si c=1 el pin 4 se pone en alto y se enciende el Led
if(c==1)
digitalWrite(4,HIGH);
// Si c=0 el pin 4 se pone en bajo y se apaga el Led
if(c==0)
digitalWrite(4,LOW);
```

```
// Si d=1 el pin 5 se pone en alto y se enciende el Led
  if(d==1)
    digitalWrite(5,HIGH);
// Si d=0 el pin 5 se pone en bajo y se apaga el Led
  if(d==0)
    digitalWrite(5,LOW);
  }}}

```

3.7.2 Aplicación automática:

Al presionar el pulsador conectado en el pin D2 (Válvula Principal) del arduino mega y se muestra en la pantalla LCD la palabra “válvula ON” cuando la entrada está en 1 lógica activándose la electroválvula y la palabra “válvula OFF” cuando la entrada está en 0 lógica desactivándose la electroválvula, al presionar el pulsador conectado en el pin D3 (secuencia) se observa en la LCD la palabra “Secuencia en curso” e inicializa la secuencia básica; “Cilindro A = ON” , “Cilindro A,B = ON”, “Cilindro A,B,C = ON”, “Cilindro A,B= ON”, “Cilindro A= ON”, de igual manera en el módulo de electroneumática se visualiza con la activación del cilindro A, cilindro B, cilindro C y después se observa la palabra “Secuencia OFF”.

3.7.2.1 Xbee Maestro Arduino Mega:

```
// Librería para la LCD
#include <LiquidCrystal.h>
// Descripción de los pines de comunicación a utilizar
//(RS,E,D4,D5,D6,D7)
LiquidCrystal lcd(12, 11, 10, 9, 8, 7);
// Declaración de variables
byte a=0,b=0,c=0,d=0;
String a1,b1,c1,d1;
byte x=0,y=0;

// creación de la funcion enviar
void enviar(byte a,byte b,byte c,byte d);

void setup()

```

```

{
// inicializa la LCD
lcd.begin(20, 4);
// Inicia la comunicación serial a una la velocidad de datos a 9600 bps
Serial.begin(9600);
// Configura el pin 2 como entrada y habilita la resistencia interna de pull-up
pinMode(2,INPUT_PULLUP);
// Configura el pin 3 como entrada y habilita la resistencia interna de pull-up
pinMode(3,INPUT_PULLUP);
}
void loop() {
// Coloca el cursor en la columna 0, la línea 0
// (Nota: la línea 0 es la primera fila, ya que el conteo se inicia en 0 )
lcd.setCursor(0, 0);
// Muestra el mensaje "Control Automatico" en la pantalla LCD
lcd.print(" Control Automatico");
// Coloca el cursor en la columna 2, la línea 0
lcd.setCursor(0, 1);
// Muestra el mensaje "Valvula" en la pantalla LCD
lcd.print("Valvula: ");

// Lee el valor del pin digital 2 especificado, ya sea alta o baja.
// Si r<1 y a=1 entonces se mostrara el mensaje válvula ON en la LCD caso
contrario si r=0 y a=0 se mostrara válvula OFF
if (!digitalRead(2))
{

if (x<1)
{
x++;
b1=" ON ";
}

else

```

```

{
x=0;
b1=" OFF";
}

a=x;
// Coloca el cursor en la columna 1, la línea 0
lcd.setCursor(0, 1);
// Muestra el mensaje Valvula en la pantalla LCD
lcd.print("Valvula: ");
// Muestra un mensaje en la pantalla LCD de la variable b1
lcd.print(b1);
// Ejecuta la función enviar
enviar(a,b,c,d);
// Retardo de 3s
delay(300);
}
// Lee el valor del pin digital 3 especificado, ya sea alta o baja.
// Se habilitara la secuencia cuando y<1 y x=1, visualizando en la LCD el
mensaje "en curso"
if (!digitalRead(3))
{
if (y<1 && x==1)
{
y++;
c1=" En curso";
// Cilindro A, en 1 y cilindro B, C en 0
d1=" A= ON";
b=1;
c=0;
d=0;
// Coloca el cursor en la columna 2, la línea 0
lcd.setCursor(0, 2);
// Muestra la palabra "Secuencia" en la pantalla LCD

```

```
    lcd.print("Secuencia: ");
// Muestra un mensaje en la pantalla LCD de la variable c1
    lcd.print(c1);
// Coloca el cursor en la columna 3, la línea 0
    lcd.setCursor(0, 3);
// Muestra la palabra "Cilindro" en la pantalla LCD
    lcd.print("Cilindro ");
// Muestra un mensaje en la pantalla LCD de la variable d1
    lcd.print(d1);
//Ejecuta el lazo "For" hasta que sea menor o igual que 300
    for(int i=0;i<=300;i++)
    {
// ejecuta la función enviar
        enviar(a,b,c,d);
// retardo de 1ms
        delay(1);
    }
    // Cilindro A, B en 1 y cilindro C en 0
    d1=" A,B= ON";
    b=1;
    c=1;
    d=0;
    // Coloca el cursor en la columna 2, la línea 0
    lcd.setCursor(0, 2);
// Muestra la palabra "Secuencia" en la pantalla LCD
    lcd.print("Secuencia: ");
// Muestra un mensaje en la pantalla LCD de la variable c1
    lcd.print(c1);
    // Coloca el cursor en la columna 3, la línea 0
    lcd.setCursor(0, 3);
// Muestra la palabra Cilindro en la pantalla LCD en la columna 3,línea 0
    lcd.print("Cilindro ");
// Muestra un mensaje en la pantalla LCD de la variable d1
    lcd.print(d1);
```

```

//Ejecuta el laso "For" hasta que sea menor o igual que 300
  for(int i=0;i<=300;i++)
  {
// Ejecuta a la función enviar
    enviar(a,b,c,d);
// retardo de 1ms
    delay(1);
  }
// Cilindro A, B, C en 1
  d1=" A,B,C= ON";
  b=1;
  c=1;
  d=1;
// Coloca el cursor en la columna 2, la línea 0
  lcd.setCursor(0, 2);
// Muestra la palabra "Secuencia" en la pantalla LCD
  lcd.print("Secuencia: ");
// Muestra un mensaje en la pantalla LCD de la variable c1
  lcd.print(c1);
// Coloca el cursor en la columna 3, la línea 0
  lcd.setCursor(0, 3);
// Muestra la palabra "Cilindro" en la pantalla LCD
  lcd.print("Cilindro ");
// Muestra un mensaje en la pantalla LCD de la variable d1
  lcd.print(d1);

//Ejecuta el laso "For" hasta que sea menor o igual que 300
  for(int i=0;i<=300;i++)
  {
//Ejecuta la función enviar
    enviar(a,b,c,d);
//Retardo de 1ms
    delay(1);
  }

```

```

// Cilindro A,B en 1 y cilindro C en 0
d1=" A,B= ON ";
b=1;
c=1;
d=0;
// Coloca el cursor en la columna 2, la línea 0
  lcd.setCursor(0, 2);
// Muestra la palabra "Secuencia" en la pantalla LCD
  lcd.print("Secuencia: ");
// Muestra un mensaje en la pantalla LCD de la variable c1
  lcd.print(c1);
// Coloca el cursor en la columna 3, la línea 0
  lcd.setCursor(0, 3);
// Muestra la palabra "Cilindro" en la pantalla LCD
  lcd.print("Cilindro ");
// Muestra un mensaje en la pantalla LCD de la variable d1
  lcd.print(d1);

//Ejecuta el laso "For" hasta que sea menor o igual que 300
  for(int i=0;i<=300;i++)
  {
//Ejecuta la función enviar
    enviar(a,b,c,d);
//Retardo de 1ms
    delay(1);
  }
// Cilindro A, en 1 y cilindro B, C en 0
d1=" A= ON ";
b=1;
c=0;
d=0;
// Coloca el cursor en la columna 2, la línea 0
  lcd.setCursor(0, 2);
// Muestra la palabra "Secuencia:" en la pantalla LCD

```

```

    lcd.print("Secuencia: ");
// Muestra un mensaje en la pantalla LCD de la variable c1
    lcd.print(c1);
// Coloca el cursor en la columna 3, la línea 0
    lcd.setCursor(0, 3);
// Muestra la palabra "Cilindro" en la pantalla LCD
    lcd.print("Cilindro ");
// Muestra un mensaje en la pantalla LCD de la variable d1
    lcd.print(d1);
//Ejecuta el lazo "For" hasta que sea menor o igual que 300
    for(int i=0;i<=300;i++)
    {
//Ejecuta la función enviar
        enviar(a,b,c,d);
//Retardo de 1ms
        delay(1);
    }
// Cilindro A,B, C en 0 y visualizaremos en la LCD el mensaje "secuencia OFF"
    b=0;
    c=0;
    d=0;
    c1=" OFF      ";
    d1="          ";
    y=0;
// Coloca el cursor en la columna 2, la línea 0
    lcd.setCursor(0, 2);
// Muestra el mensaje Secuencia en la pantalla LCD
    lcd.print("Secuencia: ");
// Muestra un mensaje en la pantalla LCD de la variable c1
    lcd.print(c1);
// Coloca el cursor en la columna 3, la línea 0
    lcd.setCursor(0, 3);
    lcd.print("      ");
// Muestra un mensaje en la pantalla LCD de la variable d1

```



```

    lcd.print(d1);
  }
}
//Ejecuta la función enviar
enviar(a,b,c,d);
// Coloca el cursor en la columna 1, la línea 0
    lcd.setCursor(0, 1);
// Muestra la palabra "Valvula:" en la pantalla LCD
    lcd.print("Valvula: ");
// Muestra un mensaje en la pantalla LCD de la variable b1
    lcd.print(b1);
// Coloca el cursor en la columna 2, la línea 0
    lcd.setCursor(0, 2);
// Muestra la palabra "Secuencia:" en la pantalla LCD
    lcd.print("Secuencia: ");
// Muestra un mensaje en la pantalla LCD de la variable c1
    lcd.print(c1);
}

```

3.7.2.2 Xbee Esclavo

```

void setup() {
//Inicia la comunicación serial y la velocidad de datos a 9600 bps
Serial.begin(9600);
//Declaracion del pin 2 como salida, activarlo aplicando 5v en el pin y
desactivarlo con 0V
pinMode(2,OUTPUT);
//Declaracion del pin 3 como salida, activarlo aplicando 5v en el pin y
desactivarlo con 0V
pinMode(3,OUTPUT);
}

void loop() {
// Enviar datos sólo cuando se reciben datos:
while(Serial.available(>0)

```

```
{  
// Devuelve el primer número entero válido ( largo) de la memoria intermedia  
serie.  
//[Los caracteres que no sean números enteros (o el signo menos ) se  
omiten .  
byte a=Serial.parseInt();  
  
// Lee datos serie entrantes  
if(Serial.read()=='n')  
{  
// Si a=1 el pin 2 se pone en alto y se enciende el Led  
if(a==1)  
digitalWrite(2,HIGH);  
// Si a=0 el pin 2 se pone en bajo y se apaga el Led  
if(a==0)  
digitalWrite(2,LOW);  
// Si b=1 el pin 3 se pone en alto y se enciende el Led  
if(b==1)  
digitalWrite(3,HIGH);  
// Si b=0 el pin 3 se pone en bajo y se apaga el Led  
if(b==0)  
digitalWrite(3,LOW);  
}}}
```

3.8 Circuito del control inalámbrico

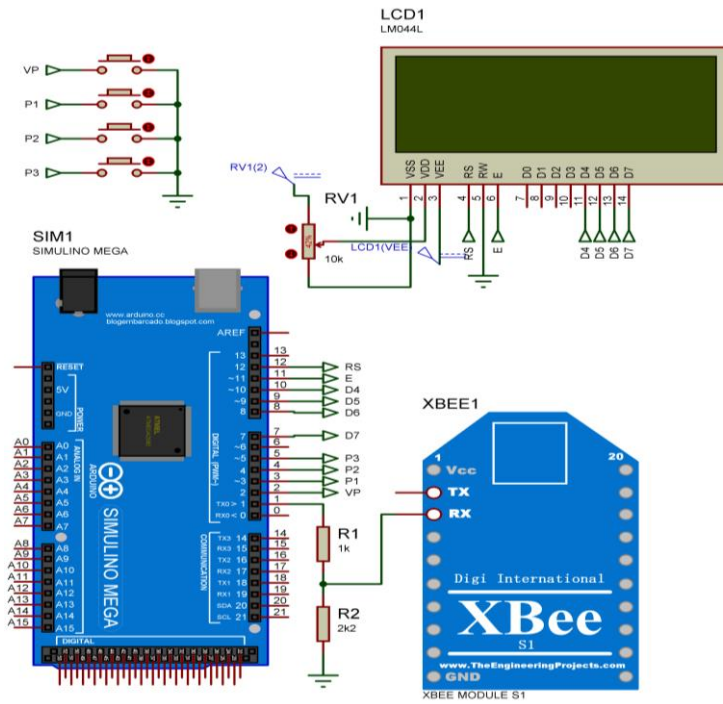


Figura 64 Circuito de Control (Maestro).

Fuente: Montaluisa Washington

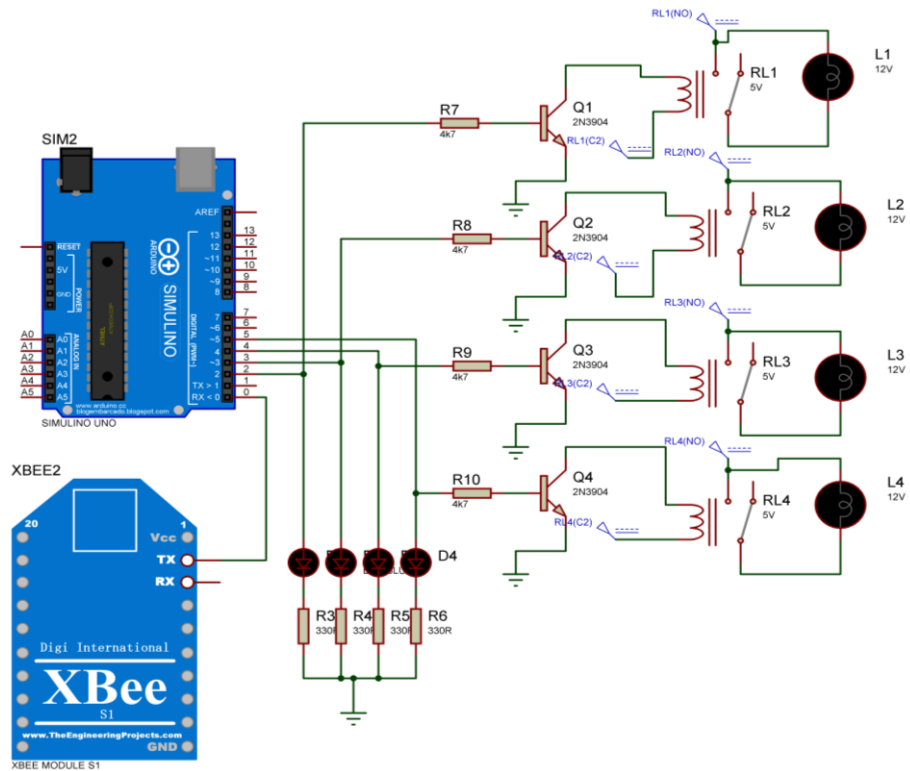


Figura 65 Circuito de Potencia (Esclavo).

Fuente: Montaluisa Washington

3.9 Esquema del circuito de potencia en Proteus

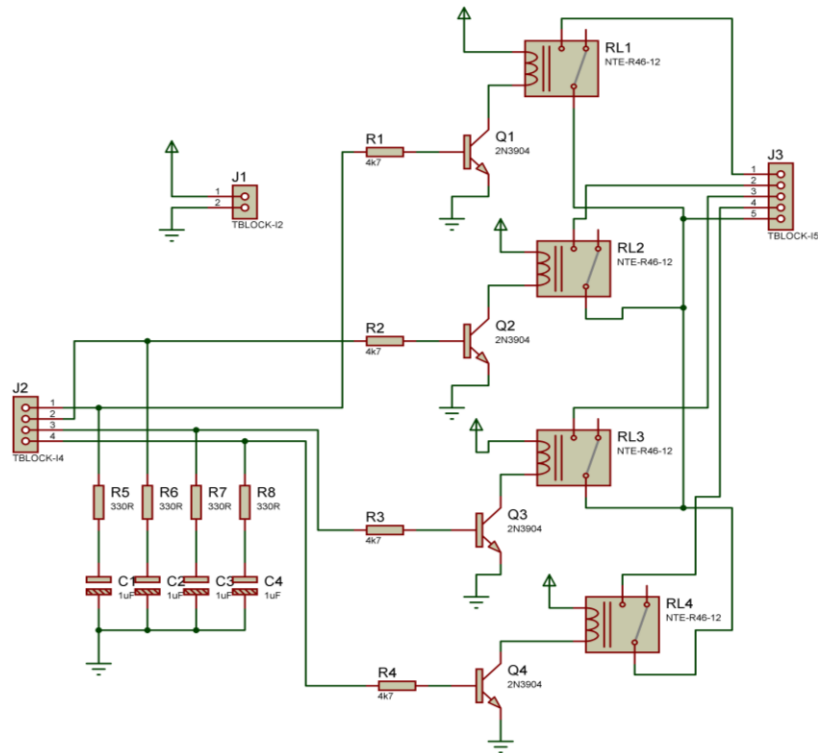


Figura 66 Esquema de potencia.

Fuente: Montaluisa Washington

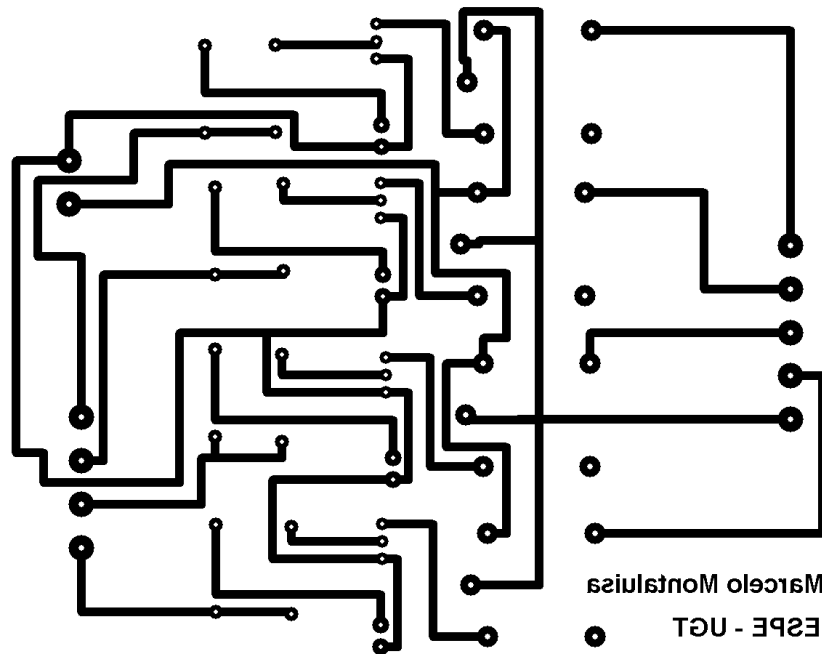


Figura 67 Circuito Diseñado para el esquema de potencia.

Fuente: Montaluisa Washington

3.10 Esquema del shield del arduino mega

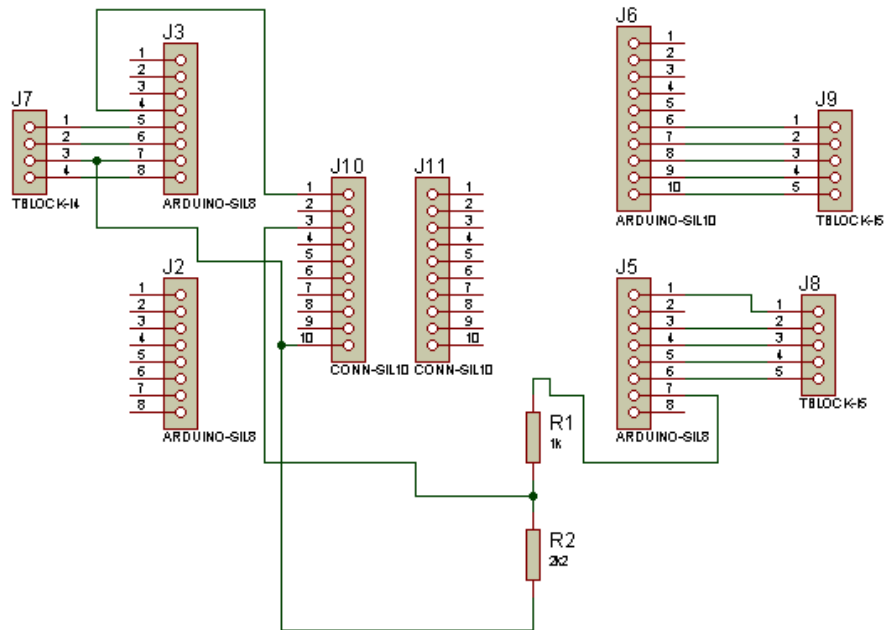


Figura 68 Shield arduino mega.

Fuente: Montaluisa Washington

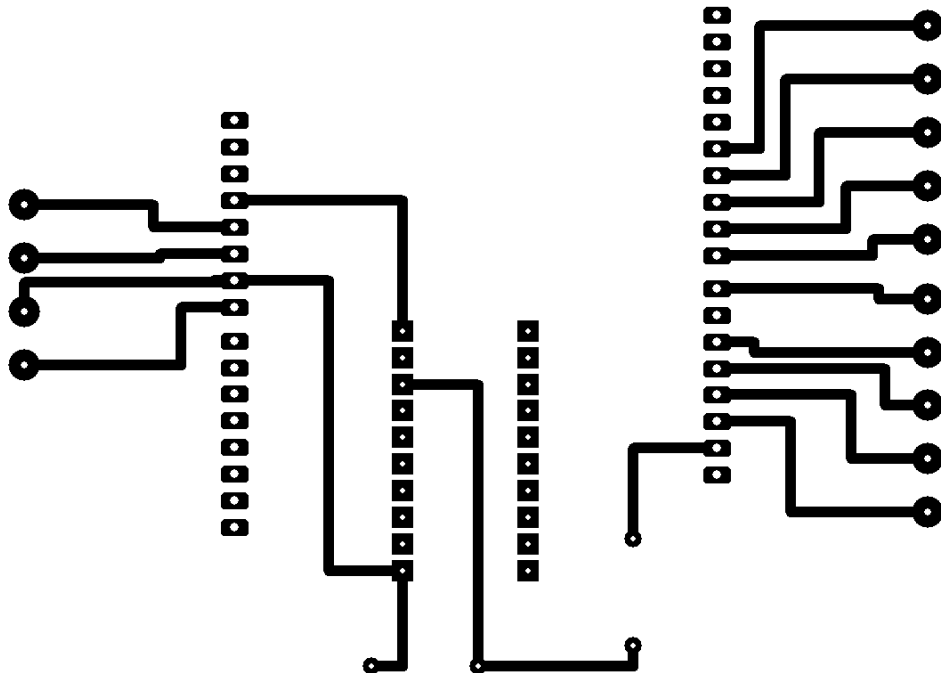


Figura 69 Esquema diseñado del shield arduino mega.

Fuente: Montaluisa Washington

3.11 Esquema del shield del arduino uno

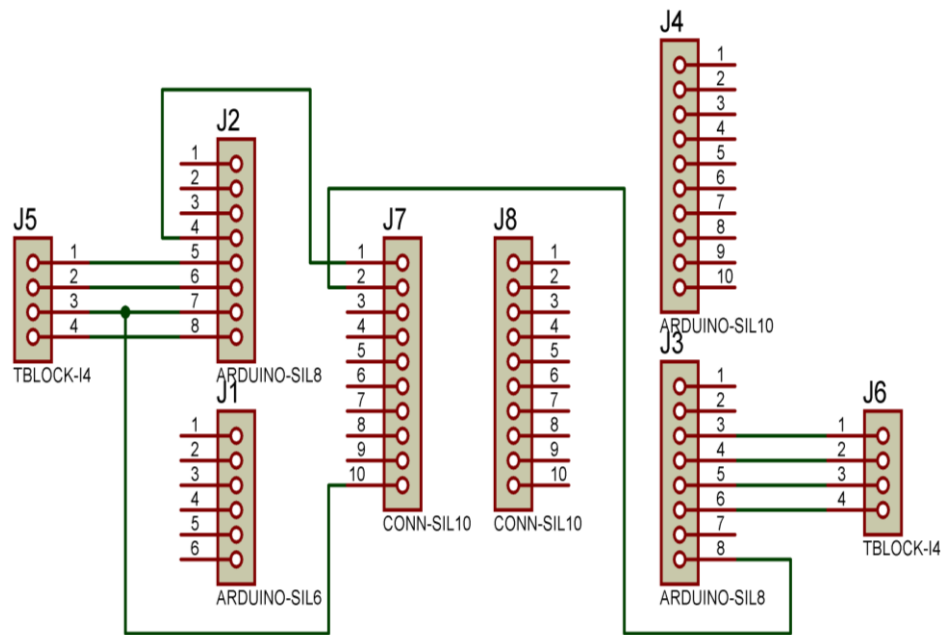


Figura 70 Shield arduino uno.

Fuente: Montaluisa Washington

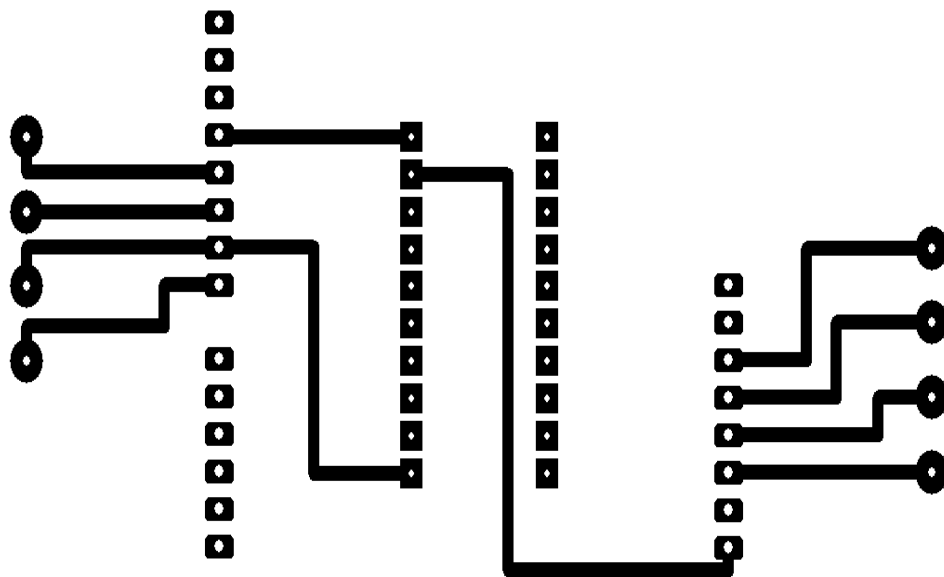


Figura 71 Esquema diseñado del shield arduino uno.

Fuente: Montaluisa Washington

3.12 Esquemas de conexión del Modulo Electroneumatica

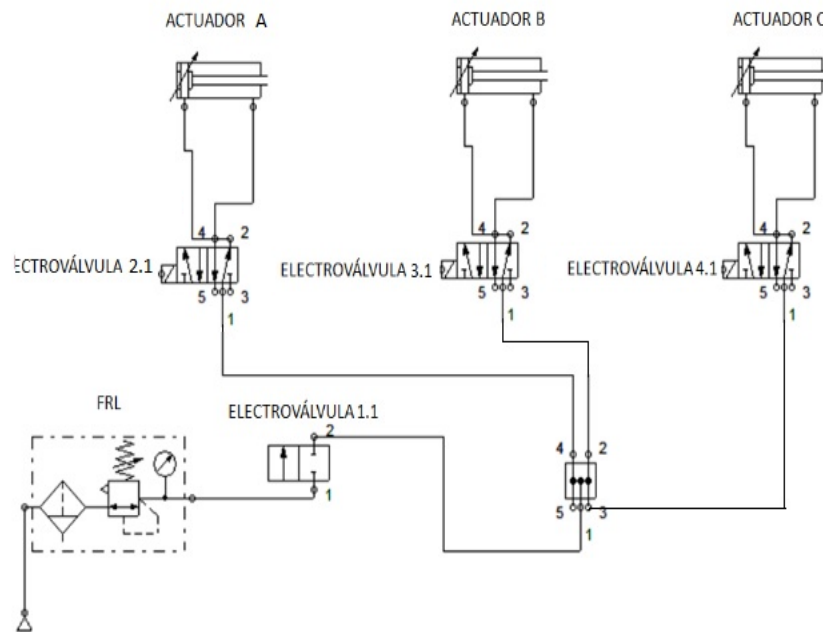


Figura 72 Esquema Neumatico.

Fuente: Montaluisa Washington

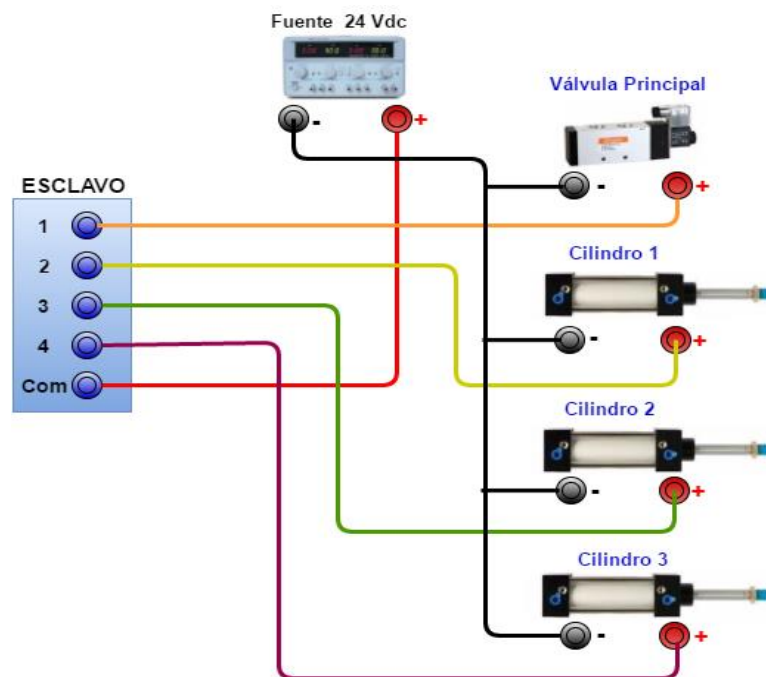


Figura 73 Esquema Electrico.

Fuente: Montaluisa Washington

3.13 Pruebas de funcionamiento

3.13.1 Monitor serie del arduino mega con la aplicación manual

En la figura 74, se observa el monitor serie con la válvula principal (VP) y el cilindro 1 (C1), cilindro 2 (C2), cilindro 3 (C3) apagados.

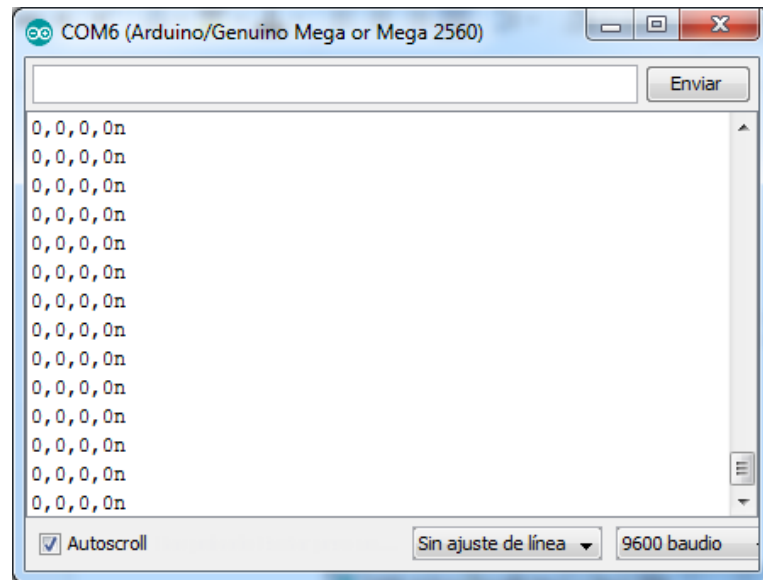


Figura 74 VP, C1, C2, C3 apagados.

Fuente: (Arduino, 2016)

En la figura 75, se observa la válvula principal activada y el cilindro 1, cilindro 2, cilindro 3 apagados.



Figura 75 VP activado, C1, C2, C3 apagados.

Fuente: (Arduino, 2016)

En la figura 76, se observa la válvula principal y el cilindro 1 activados y el cilindro 2, cilindro 3 apagados.

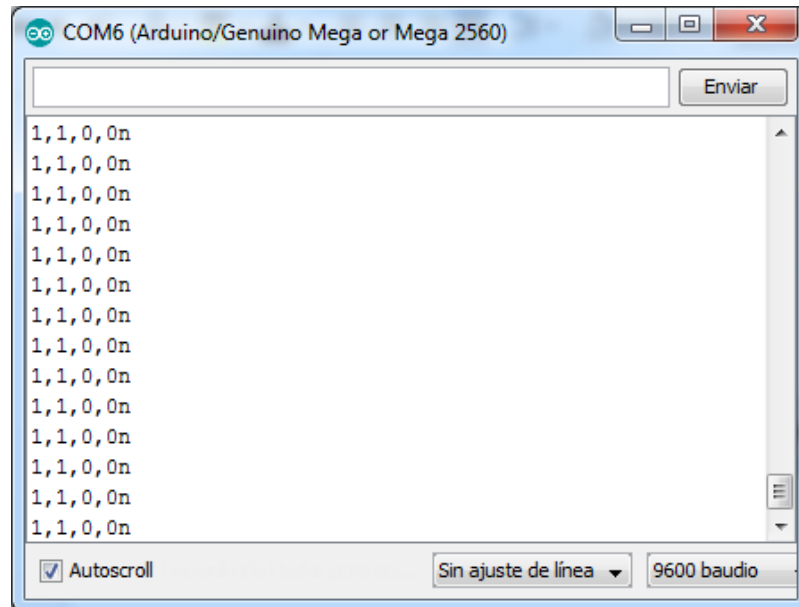


Figura 76 VP, CA activados y CB, CC apagados.

Fuente: (Arduino, 2016)

En la figura 77, se observa la válvula principal y el cilindro 1, cilindro 2 activados y el cilindro 3 apagado.



Figura 77 VP, C1, C2, activados y C3 apagado.

Fuente: (Arduino, 2016)

En la figura 78, se observa la válvula principal y el cilindro 1, cilindro 2, cilindro 3 activados.

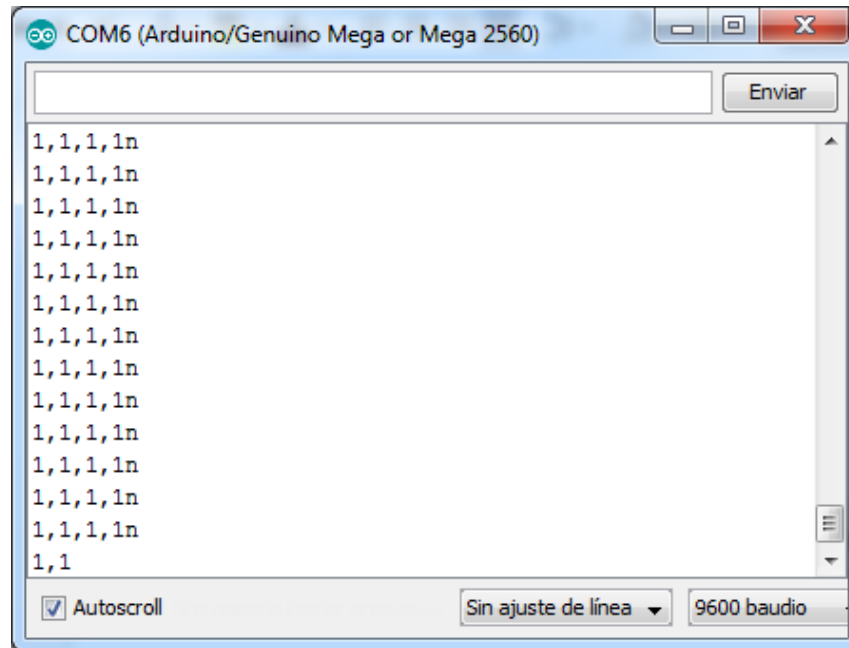


Figura 78 VP, C1, C2, C3 activadas.

Fuente: (Arduino, 2016)

3.13.2 Pruebas en el módulo de electroneumática

Se verifico que el la LCD y en el módulo de electroneumática se activen la válvula principal (VP) y cilindro 1 (C1), cilindro 2 (C2) y cilindro 3 (C3) como se indica en las siguientes figuras.

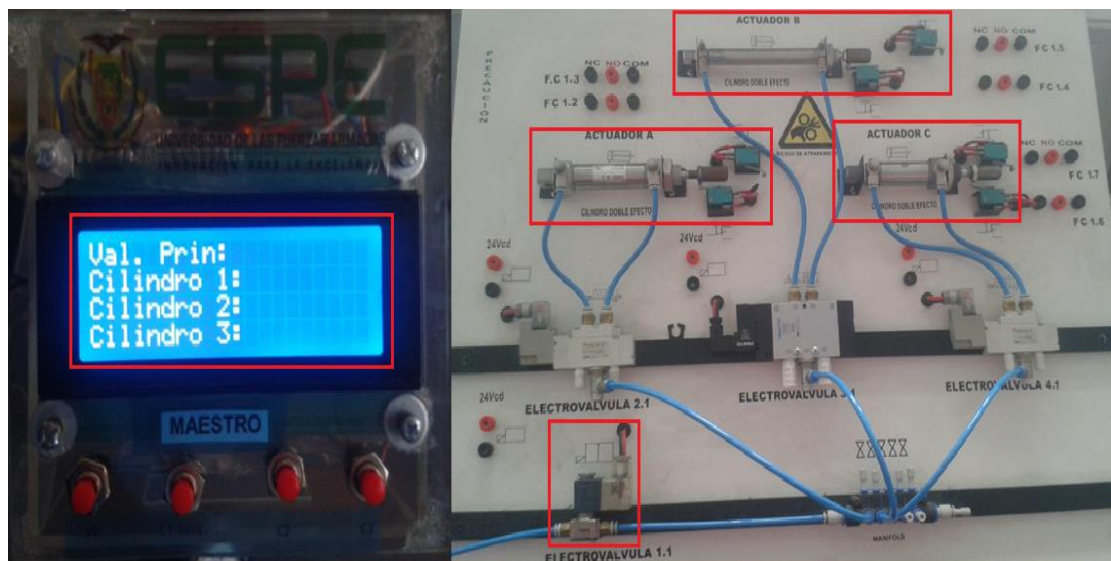


Figura 79 VP, C1, C2, C3 apagados.

Fuente: Montaluisa Washington

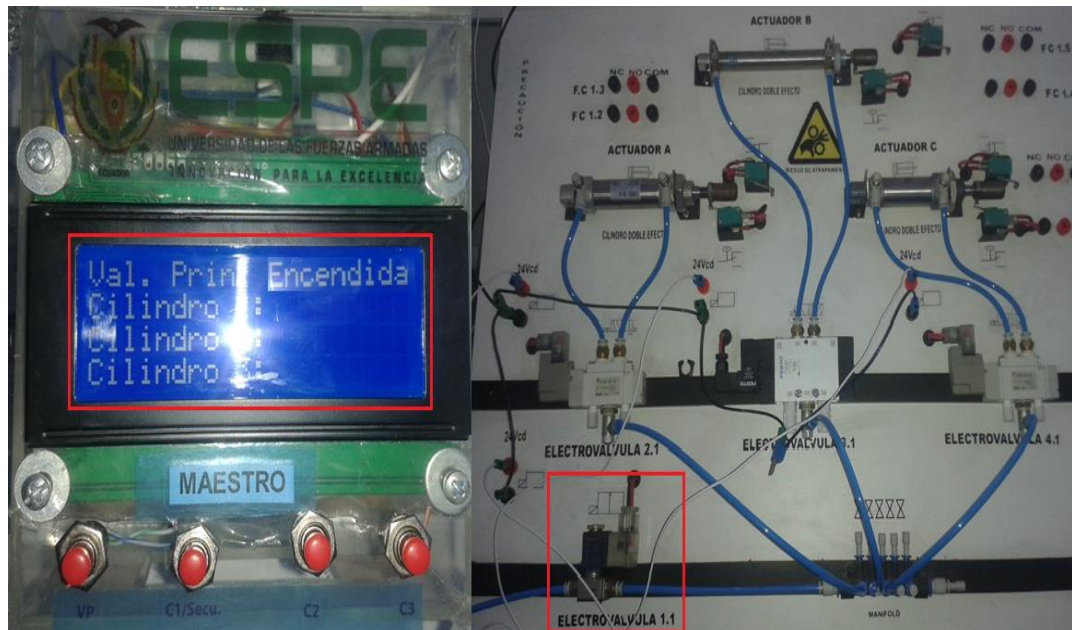


Figura 80 VP encendida, C1, C2, C3 adentro.

Fuente: Montaluisa Washington

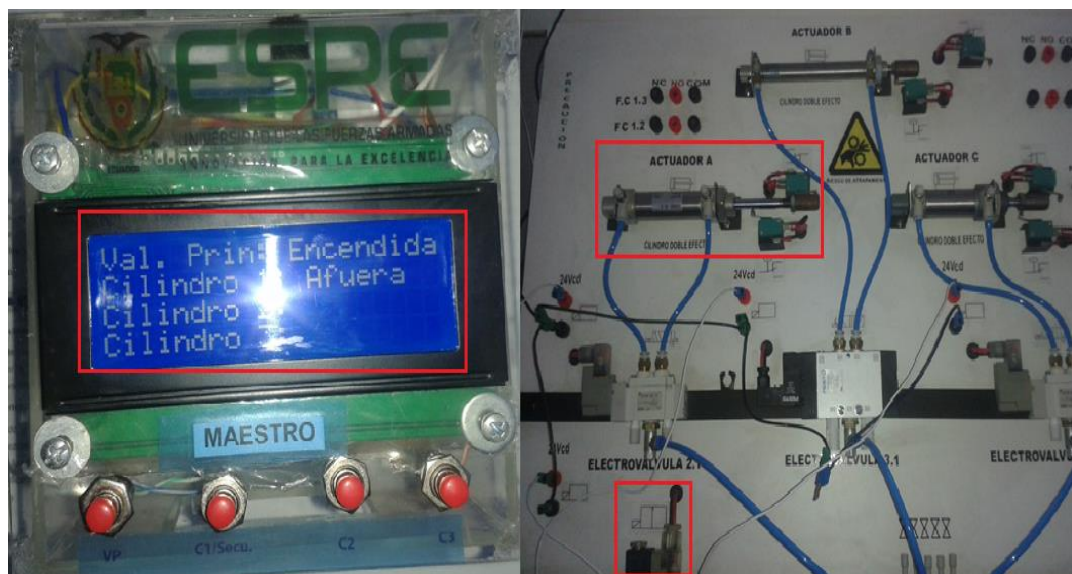


Figura 81 VP encendida, C1 afuera y C2, C3 adentro.

Fuente: Montaluisa Washington

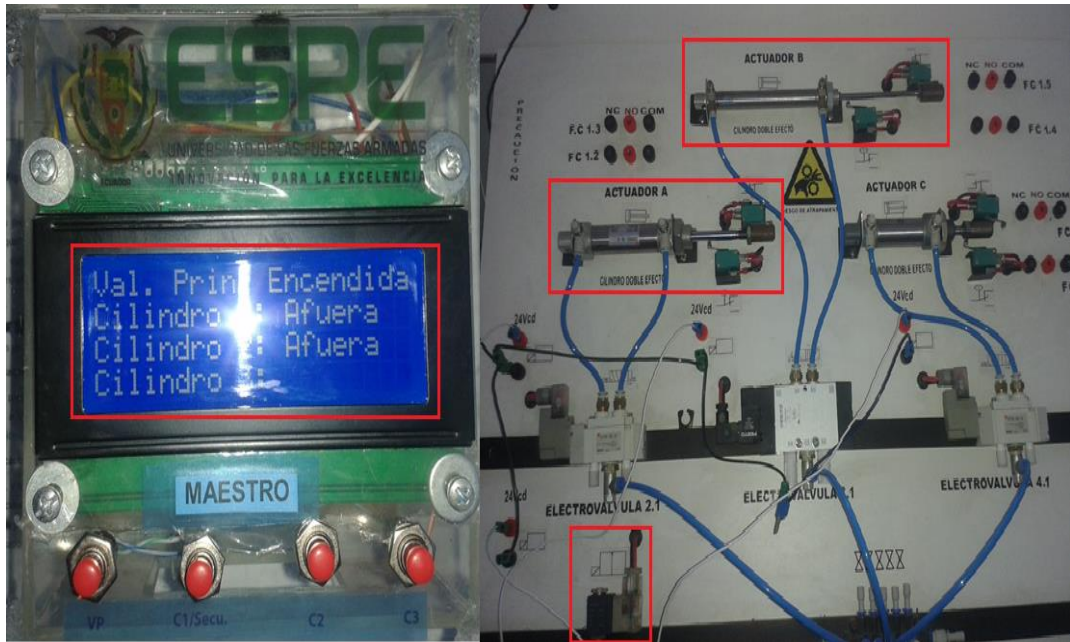


Figura 82 VP encendida, C1, C2 afuera y C3 adentro.

Fuente: Montaluisa Washington

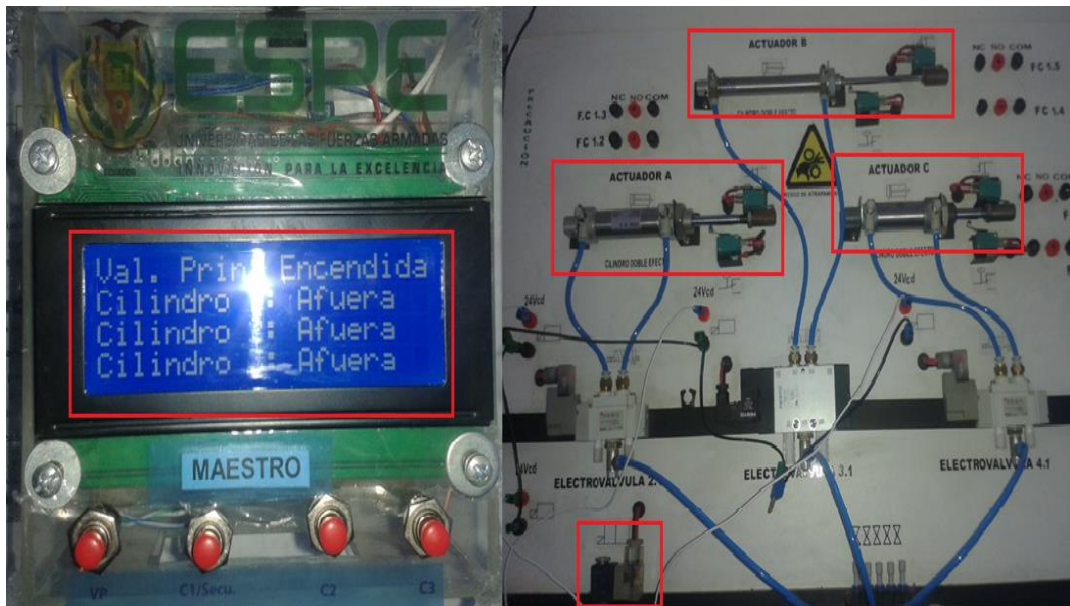


Figura 83 VP encendida, C1, C2, C3 afuera.

Fuente: Montaluisa Washington

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES.

4.1 Conclusiones.

- La investigación sobre las características y el funcionamiento del módulo Xbee Pro, permitió comprender de mejor manera todos los parámetros necesarios para un correcto funcionamiento del módulo inalámbrico.
- Los requerimientos mínimos de modulo adquirido para la implementación control , permitió realizar la comunicación inalámbrica de una manera fácil de comprender utilizando los software XCTU que permite configurar el módulo Xbee Pro y el software Arduino el cual permite realizar la programación para el control del módulo de electroneumática , lo que facilitó la elaboración del proyecto técnico.
- Se determinó las principales ventajas y los diferentes campos de aplicación del uso de la tecnología ZigBee ante las demás tecnologías como Bluetooth y WiFi, lo cual reduce costos de operación, instalación y mantenimiento.
- Se implementó una comunicación inalámbrica para el control del módulo de electroneumática en el cual se realizó las conexiones y la alimentación de acuerdo a los datos establecidos para este proyecto

técnico y el software arduino que ayudara para la trasmisión y recepción de datos entre los módulos xbee pro, los mismos que servirán para realizar prácticas relacionadas al control industrial y la domótica.

- Dentro de las pruebas de funcionamiento, el módulo Xbee Pro maestro y el módulo Xbee Pro esclavo trabajan exitosamente cumpliendo la secuencia de cilindros básica correspondiente a este proyecto técnico.

4.2 Recomendaciones.

- Revisar las especificaciones y características técnicas de los módulos Xbee Pro para realizar una correcta conexión con el arduino mega y el arduino uno antes de alimentar a los dispositivos con voltaje.
- Retirar el módulo Xbee Pro antes de subir el código fuente a la placa arduino.
- Para el uso de los módulos Xbee Pro es esencial diseñar una fuente de alimentación de corriente directa confiable y precisa, o utilizar un regulador de 3.3V para asegurar un voltaje constante, ya que los dispositivos Xbee Pro son sensibles a fallas de voltaje y pueden resultar dañados de forma permanente.
- Se debe ubicar los módulos Xbee Pro tratando que tengan la mayor línea de vista posible, para garantizar la fiabilidad de transmisión de datos.
- Crear una función para enviar datos en el código fuente del módulo maestro para que no exista latencia.

GLOSARIO DE TÉRMINOS

ADC	Conversor Análogo Digital
CH	Canal
CSMA-CA	Carrier Sense Multiple Access Collision Avoidance
DH	Dirección de destino de alta
DL	Dirección de destino bajo
GTS	Guaranteed time slot (Ranura de Tiempo Garantizado)
ID	PAN ID
IDE	Entorno de desarrollo integrado
IEEE	Instituto de Ingenieros Eléctricos, Electrónicos
FFD	Dispositivo de funcionalidad completa
KB	Kilobytes
Kbps	Kilobites por segundos
MY	Dirección de origen
PAN	Red de área personal
PHY	Capa Física
PL	Nivel de energía
RF	Radio Frecuencia
Rx	Recepción
TTL	Lógica Transistor Transistor
Tx	Transmisión
USB	Bus Universal en Serie
Vin	Voltaje de entrada

WLAN	Redes inalámbricas de área local
WMAN	Redes inalámbricas de área metropolitana
WPAN	Redes inalámbricas de área personal
WWAN	Redes inalámbricas de área geográfica

REFERENCIAS BIBLIOGRAFICAS

- Airtec. (2012). *airtec*. Obtenido de <http://www.airtec.de/mechanically-and-manually-operated-valves.html>
- Altec. (2015). Obtenido de <http://www.altecdust.com/soporte-tecnico/que-son-las-electrovalvulas>
- Alvarez, I. (Febrero de 2014). *Batraco*. Obtenido de <http://batrako.blogspot.com/2013/02/posibilidades-del-arduino-en-simracing.html>
- Andi, L. (2012). Visualización de señales de velocidad y presión desde el mini UAV mediante Xbee Pro 900. *Trabajo de graduación*. Latacunga, Cotopaxi, Ecuador: ESPE-ITSA.
- Arduino. (Marzo de 2015). Obtenido de Arduino cc
- Arduino. (2016). Obtenido de Arduino.cc: <https://www.arduino.cc/en/Main/Software>
- Arduino. (2016). Obtenido de <https://www.arduino.cc/en/main/arduinoBoardUno>
- Arduino. (2016). *Arduino mega*. Obtenido de arduino.cc: <https://www.arduino.cc/en/Main/arduinoBoardMega>
- Arduino. (2016). *Arduino Uno*. Obtenido de <https://www.arduino.cc/en/main/arduinoBoardUno>
- Ariston. (2013). *ARISTON*. Obtenido de <https://blog.ariston.es/2014/06/21/5-sencillos-proyectos-para-desarrollar-con-arduino-para-este-verano/>
- Atmels. (2 de Enero de 2016). *instructables*. Obtenido de <http://www.instructables.com/id/Arduino-on-all-sorts-of-Atmels/>

- Blogger. (28 de Abril de 2011). *electroneumatic.blogspot*. Obtenido de <http://electroneumatic.blogspot.com/2011/04/electroneumatica-basica.html>
- Bueno. (2013). *portaleso.com*. Obtenido de http://www.portaleso.com/usuarios/Toni/web_neumatica/neumatica_in_dice.html
- Cabrera, E. (2012). *monografias.com*. Obtenido de monografias.com: <http://www.monografias.com/trabajos14/control/control.shtml>
- Commerce, L. (2016). Obtenido de <http://www.lck-led.com/arduino-mega-2560-microcontroller-board-atmega16u2-free-p-1169.html?cPath=151>
- Digi. (2016). Obtenido de www.digi.com/support/eservice
- Digi. (enero de 2016). Obtenido de http://www.digi.com/resources/documentation/digidocs/90001458-13/default.htm#concept/c_xctu_layout.htm%3FTocPath%3D_____4
- Digi. (ENERO de 2016). *DIGI*. Obtenido de <http://www.digi.com/products/xbee-rf-solutions/xctu-software/xctu>
- Digi. (2016). *Digi International Inc.* Obtenido de <http://www.digi.com/products/xbee-rf-solutions/xctu-software/xctu>
- Digi. (2016). *DIGI_support*. Obtenido de <http://www.digi.com/support/productdetail?pid=3238&type=drivers>
- Digikey. (2016). *digikey.com*. Obtenido de <http://www.digikey.com/product-detail/en/digi-international/XBIB-U/XBIB-U-ND/935972>
- Dignani, J. P. (Junio de 2011). Analisis del Protocolo Zigbee. *Analisis del Protocolo Zigbee*. Buenos Aires, Argentina.
- Duerte, A. (2014). *Andres Duerte*. Obtenido de <http://www.andresduarte.com/arduino-y-xbee>
- Ecuared. (2015). *Ecuared*. Obtenido de Ecuared: http://www.ecured.cu/Topolog%C3%ADas_de_red_ZigBee
- Festo. (2015). Obtenido de ELECTROVALVULA-FESTO: <http://www.suministros-industriales-ferreteria.com/ES/articulo/ELECTROVALVULA-FESTO-CPE18-M1H-3GL-1-4/3A7FE101F5B9936C-C8C1DBDA7C1C6767.html>
- Glen, M. (23 de Mayo de 2012). *wikispaces*. Obtenido de wikispaces: <https://sx-de-tx.wikispaces.com/ZIGBEE>

- Gonzales, F. (23 de Agosto de 2014). *Dewkdd*. Obtenido de <http://dewkdd.blogspot.com/2014/08/cilindros-neumaticos.html>
- leespain. (2012). *iee*. Obtenido de <http://www.ieespain.com/ieeproteus/>
- Isaac_PE. (2014). Obtenido de <http://comohacer.eu/analisis-comparativo-placas-arduino-oficiales-compatibles/>
- Jara, C. (2008). *Definicion.de*. Obtenido de Definicion.de: <http://definicion.de/control/>
- Jimenez. (26 de Septiembre de 2011). *blogspot*. Obtenido de blogspot: <http://tecnologicobj12.blogspot.com/2011/09/que-es-xbee.html>
- Londoño, J. D. (17 de Abril de 2013). *prezi.com*. Obtenido de https://prezi.com/gg1_daqx1sha/zegbee/
- Microautomacion. (2014). *microautomacion.com*. Obtenido de <http://www.microautomacion.com/catalogo/02Valvulas.pdf>
- Parreño. (Octubre de 2012). *DISEÑO Y CONSTRUCCIÓN DE UN BANCO DE PRUEBAS DE CONTROL NEUMATICO, CON TOUCH PANEL Y S7-1200*. Latacunga, Cotopaxi, Ecuador: Universidad Tecnica de Cotopaxi.
- Proteus V8.1. (2015). *Proteus V8.1*. Labcenter Electronics.
- Saramago. (2016). *sites.google*. Obtenido de <https://sites.google.com/site/migueltecnologia/4o-eso/teoria/tema-7-neumatica-e-hidraulica>
- Souza. (2014). *embarcados.com*. Obtenido de <http://www.embarcados.com.br/arduino-mega-2560/>
- Tangient. (2016). *seritiumneumatica.wikispaces.com*. Obtenido de <http://seritiumneumatica.wikispaces.com/Unidad+de+mantenimiento>.
- Teloco. (2013). *Teleco*. Obtenido de http://www.teleco.com.br/es/tutoriais/es_tutorialzigbee/pagina_2.asp
- Tymkers. (15 de Febrero de 2015). Obtenido de <http://tymkrs.tumblr.com/post/17661586353/xbee-adapter-board-pin-connections>
- Weebly. (2015). *rduinodhtics*. Obtenido de rduinodhtics: <http://arduinohtics.weebly.com/>
- Xbee. (2015). Obtenido de <http://xbee.cl/xbee-pro-60mw-wire-antenna/>

Xbee. (Noviembre de 2015). Obtenido de <http://xbee.cl/xbee-pro-60mw-wire-antenna/>

XCTU, S. (Enero de 2016). *XCTU*. Obtenido de <http://www.digi.com/hottag?ht=/xctu-windows>

Zambrano, H. (marzo de 2015). Diseño e implementación de un sistema de control y monitoreo mediante LabVIEW. *Diseño e implementación de un sistema de control y monitoreo mediante LabVIEW para una plataforma giratoria aplicada a los enlaces terrestres vía microonda y a una antena de recepción satelital*. Guayaquil, Guayas, Ecuador: UPS.

DATOS PERSONALES



Nombre: MONTALUISA MONTALUISA WASHINGTON
MARCELO

Nacionalidad: Ecuatoriana

Fecha De Nacimiento: 22 DE NOVIEMBRE DE 1990

Cedula de Ciudadanía: 0502885981

Teléfonos: 0984781543/032807224

Correo Electrónico: wmmontaluisa@gmail.com

Dirección Domicilio: LATACUNGA BARRRIO SAN SEBASTIAN
(CALIXTO PINO Y NAPO)

ESTUDIOS REALIZADOS:

Primaria:

- Escuela Fiscal “Simón Bolívar” - Latacunga

Secundaria:

- Instituto Tecnológico Superior “Vicente León” - Latacunga

Superior:

- Unidad de Gestión de Tecnologías “ESPE”

TITULOS OBTENIDOS:

- Bachiller en Ciencias “Física y Matemáticas”
- Suficiencia en el idioma ingles

EXPERIENCIAS LABORALES:

- IMHOTEP CONSTRUCTORES Pasante – Técnico Electrónico
- CNT-COTOPAXI Pasante – Técnico Electrónico

ACEPTACIÓN DEL USUARIO

Latacunga, 04 de julio de 2016

Yo, ING PABLO PILATÁSIG en calidad de encargado del Laboratorio de Instrumentación Virtual de la Unidad de Gestión de Tecnologías, me permito informar lo siguiente:

El proyecto técnico elaborado por el Sr. **MONTALUISA MONTALUISA WASHINGTON MARCELO**, con el tema: “**IMPLEMENTACIÓN DE UN CONTROL INALÁMBRICO PARA EL MÓDULO DE ELECTRONEUMÁTICA MEDIANTE XBEE PRO**”, ha sido efectuado de forma satisfactoria en las dependencias de mi cargo y que la misma cuenta con todas las garantías de funcionamiento, por lo cual extiendo este aval que respalda el trabajo realizado por el mencionado estudiante.

Por tanto me hago cargo de todas las instalaciones realizadas por el Sr. estudiante.

Atentamente,

ING. PABLO PILATÁSIG

ENCARGADO DEL LABORATORIO DE INSTRUMENTACIÓN VIRTUAL

Latacunga, 22 de julio de 2016

HOJA DE LEGALIZACIÓN DE FIRMAS

**DEL CONTENIDO DE LA PRESENTE INVESTIGACIÓN SE
RESPONSABILIZA EL AUTOR**

**MONTALUISA MONTALUISA WASHINGTON MARCELO
ID L00246396**

**DIRECTOR DE LA CARRERA DE ELECTRÓNICA MENCIÓN
INSTRUMENTACIÓN & AVIÓNICA**

**Ing. Pablo Pilatásig Director Carrera de Electrónica Mención
Instrumentación & Aviónica**

