



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería Mecánica Eléctrica

**DISEÑO DE DISPOSITIVO *BLUETOOTH* PARA LA ADQUISICIÓN Y
TRANSMISIÓN DE DATOS EN EL CAMPO INDUSTRIAL**

José Miguel Ruano Aguilar

Asesorado por el Ing. Carlos Eduardo Guzmán Salazar

Guatemala, marzo de 2016

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DISEÑO DE DISPOSITIVO *BLUETOOTH* PARA LA ADQUISICIÓN Y
TRANSMISIÓN DE DATOS EN EL CAMPO INDUSTRIAL**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

JOSÉ MIGUEL RUANO AGUILAR

ASESORADO POR EL ING. CARLOS EDUARDO GUZMÁN SALAZAR

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN ELECTRÓNICA

GUATEMALA, MARZO DE 2016

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Pedro Antonio Aguilar Polanco
VOCAL I	Ing. Angel Roberto Sic García
VOCAL II	Ing. Pablo Christian de León Rodríguez
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Raúl Eduardo Ticún Córdova
VOCAL V	Br. Henry Fernando Duarte García
SECRETARIA	Inga. Lesbia Magalí Herrera López

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

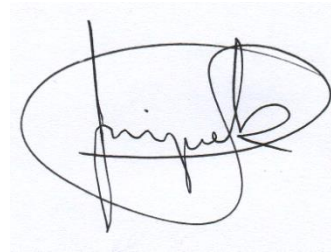
DECANO	Ing. Murphy Olympto Paiz Recinos
EXAMINADOR	Ing. Marvin Marino Hernández Fernández
EXAMINADORA	Inga. Ingrid Salomé Rodríguez de Loukota
EXAMINADORA	Inga. María Magdalena Puente Romero
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

DISEÑO DE DISPOSITIVO *BLUETOOTH* PARA LA ADQUISICIÓN Y TRANSMISIÓN DE DATOS EN EL CAMPO INDUSTRIAL

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería Mecánica Eléctrica, con fecha octubre 2013.

A handwritten signature in black ink, appearing to read 'José Miguel Ruano Aguilar', enclosed within a light blue rectangular border.

José Miguel Ruano Aguilar

Guatemala, 9 de octubre de 2015

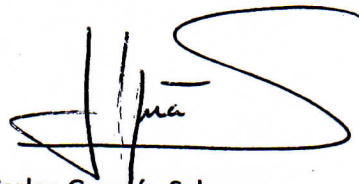
Señor
Coordinador Área de Electrónica
Escuela de Ingeniería Mecánica Eléctrica
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

Estimado Coordinador:

Hago de su conocimiento por este medio que he concluido la revisión del trabajo de graduación del estudiante **José Miguel Ruano Aguilar**, con carné 2009-14909, titulado **DISEÑO DE DISPOSITIVO BLUETOOTH PARA LA ADQUISICIÓN Y TRANSMISIÓN DE DATOS EN EL CAMPO INDUSTRIAL**. El trabajo ha cumplido con los objetivos que se propusieron, por lo que considero puede continuar con el trámite que la Universidad tiene para estos trabajos.

Así mismo, manifiesto que tanto el señor Ruano Aguilar como el suscrito, somos responsables del contenido del trabajo de graduación referido.

Un cordial saludo,



Carlos Guzmán Salazar
Asesor Nombrado
Ingeniero Electricista
Colegiado No. 2762

CARLOS GUZMAN SALAZAR
Ingeniero Electricista
Col. No. 2762



REF. EIME 06.2016.
Guatemala, 12 de OCTUBRE 2015..

FACULTAD DE INGENIERIA

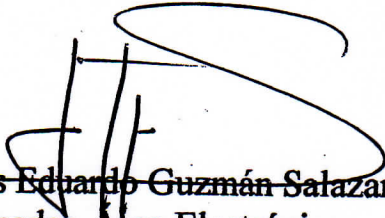
Señor Director
Ing. Francisco Javier González López
Director Escuela de Ingeniería Mecánica Eléctrica
Facultad de Ingeniería, USAC.

Señor Director:

Me permito dar aprobación al trabajo de Graduación titulado:
DISEÑO DE DISPOSITIVO BLUETOOTH PARA LA
ADQUISICIÓN Y TRANSMISIÓN DE DATOS EN EL CAMPO
INDUSTRIAL, del estudiante José Miguel Ruano Aguilar, que cumple
con los requisitos establecidos para tal fin.

Sin otro particular, aprovecho la oportunidad para saludarle.

Atentamente,
DISEÑAD Y ENSEÑAD A TODOS


~~Ing. Carlos Eduardo Guzmán Salazar~~
Coordinador Área Electrónica



SRO



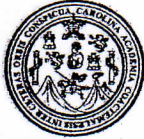
REF. EIME 06. 2016.

El Director de la Escuela de Ingeniería Mecánica Eléctrica, después de conocer el dictamen del Asesor, con el Visto bueno del Coordinador de Área, al trabajo de Graduación del estudiante; JOSÉ MIGUEL RUANO AGUILAR Titulado: DISEÑO DE DISPOSITIVO BLUETOOTH PARA LA ADQUISICIÓN Y TRANSMISIÓN DE DATOS EN EL CAMPO INDUSTRIAL, procede a la autorización del mismo.

Ing. Francisco Javier González López



GUATEMALA, 4 DE FEBRERO 2016.



El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería Mecánica Eléctrica al trabajo de graduación titulado: **DISEÑO DE DISPOSITIVO BLUETOOTH PARA LA ADQUISICIÓN Y TRANSMISIÓN DE DATOS EN EL CAMPO INDUSTRIAL**, presentado por el estudiante universitario **Jose Miguel Ruano Aguilar** y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.

Ing. Pedro Antonio Aguilar Polanco
Decano

Guatemala, marzo de 2016



ACTO QUE DEDICO A:

- Dios** Por ser mi guía en los momentos más importantes de mi vida, por darme las fuerzas cada día para salir adelante y por haberme brindado la familia que tengo.
- Mis padres** Heber Ruano y Eugenia Aguilar, personas únicas, que sin su esfuerzo y apoyo incondicional no hubiera logrado esta meta.
- Mi esposa** Virginia Ochoa, que con su amor día a día, me da la fuerza necesaria para luchar por mis sueños. Gracias por tu amor, motivación y comprensión en todo momento.
- Mi hija** Ariana Ruano, por su amor y cariño.
- Mi hermana** Alicia Ruano, por su apoyo y cariño.
- Mi abuela** Cándida Cifuentes, que en paz descansa, por sus enseñanzas, cariño, oraciones y palabras de aliento.

AGRADECIMIENTOS A:

- | | |
|-------------------------------|---|
| Facultad de Ingeniería | Por brindarme los conocimientos que me llevaron a una superación personal y por haber hecho de mí un profesional. |
| Mis amigos de Facultad | Que de una u otra manera me apoyaron para seguir adelante. Especialmente a Jorge Top, Diego Dávila, Silvio Urizar, Axel Arellano y Carlos Cabrera. Gracias por su amistad y por todos los momentos vividos. |
| Mi asesor | Ing. Carlos Guzmán, por su tiempo y apoyo. |

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	V
LISTA DE SÍMBOLOS	VII
GLOSARIO	IX
RESUMEN	XXI
OBJETIVOS.....	XXIII
INTRODUCCIÓN.....	XXV
1. MARCO TEÓRICO	1
1.1. Antecedentes	1
1.2. Concepto de <i>bluetooth</i>	1
1.3. El SIG.....	3
1.4. El mercado de <i>bluetooth</i>	4
1.5. Las actuales aplicaciones del <i>bluetooth</i>	5
1.6. Norma IEEE 802.11	7
2. COMUNICACIÓN SERIAL.....	9
2.1. Definición	9
2.1.1. Velocidad de transmisión (<i>baud rate</i>).....	10
2.1.2. Bits de datos	10
2.1.3. Bits de parada.....	11
2.1.4. Paridad.....	11
2.2. La comunicación serial en tecnología <i>bluetooth</i>	12
2.2.1. Protocolo RFCOMM	13
2.2.1.1. Circuitos que emula el RFCOMM	13

3.	PLANTEAMIENTO DEL DISPOSITIVO	15
3.1.	Generalidades del diseño	15
3.1.1.	Especificaciones del dispositivo	15
3.1.1.1.	Voltaje de operación.....	15
3.1.1.1.1.	Entrada exclusiva 5 VDC	16
3.1.1.1.2.	Entrada de 8 a 12 VDC	16
3.1.1.2.	Módulo de comunicación <i>bluetooth</i> , HC-06	16
3.1.1.2.1.	Pin KEY.....	18
3.1.1.2.2.	Pin VCC	18
3.1.1.2.3.	Pin GND.....	19
3.1.1.2.4.	Pin TXD.....	19
3.1.1.2.5.	Pin RXD	19
3.1.1.2.6.	Pin STATE	19
3.1.1.3.	Generalidades del control del dispositivo.....	20
3.1.1.3.1.	Velocidad de operación	20
3.1.1.3.2.	Entradas/salidas	20
3.1.1.3.3.	Otras especificaciones del microcontrolador	21
3.2.	Diagrama de bloques del dispositivo	22
4.	DISEÑO DE HARDWARE	23
4.1.	Control del dispositivo.....	23
4.1.1.	El PIC16F690 como control de dispositivo.....	23

4.1.2.	Reloj del microcontrolador	24
4.2.	Fuente de alimentación del dispositivo.....	25
4.3.	Entradas digitales	26
4.4.	Salidas digitales	28
4.4.1.	Relevadores de salida	28
4.4.1.1.	Circuito de excitación de bobinas de relevadores.....	30
4.5.	Listado de componentes del dispositivo.....	32
4.6.	Circuito final del dispositivo.....	33
4.7.	Diseño de circuito impreso del dispositivo.....	34
5.	PROGRAMA CONTROLADOR DEL DISPOSITIVO.....	37
5.1.	Algoritmo del programa de control.....	37
5.1.1.	Inicio.....	38
5.1.2.	Declaración de variables	38
5.1.3.	Etiquetas de programa principal.....	38
5.1.4.	Verificación de entrada de dato en puerto serial.....	38
5.1.5.	Lectura de datos en puerto.....	39
5.1.6.	Accionamiento de bandera de confirmación de lectura realizada.....	39
5.1.7.	Verificación de bandera accionada de lectura realizada	39
5.1.8.	Actividad a realizar en función al dato leído	39
5.1.8.1.	Actividades según dato entrante.....	39
5.1.9.	Desactivación de bandera de lectura realizada	40
5.1.10.	Enviar dato de entradas digitales	40
5.1.11.	Representación gráfica del algoritmo	41
5.2.	Programa de control desarrollado en MikroBasic Pro para PIC	42

5.2.1.	Aspectos importantes del compilador	42
5.2.2.	Código del programa desarrollado para el dispositivo <i>bluetooth</i>	43
5.3.	Carga de programa en microcontrolador PIC16F690	44
5.3.1.	Archivo hexadecimal.....	44
5.3.2.	Carga de código hexadecimal en la memoria del programa del microcontrolador.....	45
CONCLUSIONES.....		47
RECOMENDACIONES		49
BIBLIOGRAFÍA.....		51

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Módulo HC-06.....	16
2.	Diagrama de conexión del HC-06	17
3.	Diagrama de pines del PIC16F690	21
4.	Diagrama de bloques del dispositivo.....	22
5.	DIP20 PIC16F690.....	24
6.	Conexión de oscilador externo 1	25
7.	Conexión de oscilador externo 2	26
8.	Entradas digitales	27
9.	Medidas físicas del relevador RH-005C	29
10.	Transistor 2N3904 NPN.....	30
11.	Circuito de salida digital por relé	31
12.	Circuito final del dispositivo	34
13.	Circuito impreso, vista real	35
14.	Circuito impreso, vista de pistas área de soldadura.....	35
15.	Algoritmo del programa de control	41
16.	Código programa de control desarrollado en MikroBasic	43
17.	Código de archivo hexadecimal	44
18.	Programador QL2006.....	45

TABLAS

I.	Circuitos emulados con RFCOMM	13
II.	Pines del HC-06.....	18

III.	Especificaciones de entradas, salidas y memoria.....	22
IV.	Especificaciones de entradas, salidas y memoria.....	27
V.	Nomenclatura de pines del transistor.....	31
VI.	Pines habilitadores de salidas digitales.....	32
VII.	Listado de componentes del dispositivo.....	33
VIII.	Actividad a realizar por el microcontrolador en función al dato de entrada esperando en el puerto serial.....	40

LISTA DE SÍMBOLOS

Símbolo	Significado
GHz	Gigahercio
Kbps	Kilobits por segundo
Mbps	Megabits por segundo
MHz	Megahercio
mA	Miliamperio
mW	Milivatio

GLOSARIO

Algoritmo	Conjunto ordenado de operaciones sistemáticas que permite hacer un cálculo y hallar la solución de un tipo de problemas.
Android	Sistema operativo diseñado para dispositivos móviles, basado en el núcleo Linux.
Automatización	Sistema donde se transfieren tareas de producción, realizadas habitualmente por operadores humanos, a un conjunto de elementos tecnológicos.
<i>Baud-rate</i>	Número de unidades de señal por segundo, es también conocida como baudaje. Se le llama también velocidad de transmisión de bits por segundo.
Baudio	Unidad de medida utilizada en telecomunicaciones, que representa el número de símbolos por segundo en un medio de transmisión digital.
Bidireccional	Que tiene dos direcciones. Que puede ir en dos vías.
Bit	Señal electrónica, la cual puede encontrarse únicamente en dos estados. También llamado digito binario.

Bluetooth	Estándar de facto concebido para establecer conexiones sin hilos de corto alcance entre un amplio número de dispositivos electrónicos, fijos y móviles.
Bobina	Componente eléctrico, arrollado de hilo de cobre aislado, con diámetro y largo definido, variable según su uso.
Bornera	Conjunto de bornes individuales o múltiples y sus respectivos accesorios, utilizados para conexión de cableado eléctricos a tableros de automatización.
Byte	Conjunto de 8 bits que recibe el tratamiento de una unidad y que constituye el mínimo elemento de memoria direccionable de una computadora.
CDMA	Multiplexación por división de código (en inglés, <i>code division multiple access</i>), es un término genérico para varios métodos de multiplexación o control de acceso al medio, basados en la tecnología de espectro expandido.
Circuitería	Todos aquellos elementos eléctricos o electrónicos que constituyen un circuito que fue diseñado para realizar una función específica.

Circuito	Red eléctrica constituida por dos o más componentes, tales como resistencias, inductores, condensadores, fuentes, interruptores y demás elementos eléctricos y/o electrónicos.
CMOS	Semiconductor complementario de óxido metálico (en inglés, <i>complementary metal-oxide-semiconductor</i>), es una de las familias lógicas empleadas en la fabricación de circuitos integrados.
Comandos AT	Lenguaje desarrollado por Hayes Communications, el cual se convirtió en estándar abierto de comandos para configurar y parametrizar módems.
Compilador	Programa informático que se encarga de traducir el código fuente de una aplicación que esté en desarrollo a lenguaje de máquina. Es decir, convierte un programa realizado en lenguaje de alto nivel y lo transforma a lenguaje de bajo nivel.
Comunicación serie	Protocolo común para comunicación entre dispositivos que se incluye de manera estándar en prácticamente cualquier computadora. Comunicación que envía datos un bit a la vez.
DIP	Siglas en inglés de <i>dual in-line package</i> , es una forma de encapsulamiento, muy común en la construcción de circuitos integrados.

DUN	Siglas en inglés de <i>dual-up networking profile</i> (conexión a red por línea conmutada), permite un funcionamiento razonablemente sencillo con productos existentes, por su similitud con los protocolos de línea serie.
Emular	Imitar una cosa o a una persona procurando igualarla o incluso mejorarla. En informática es un software capaz de ejecutar programas en una plataforma diferente de la cual fue creado.
Estándar	Referencia a lo que es establecido por la autoridad, la costumbre o el consentimiento general.
Extensión	Cadena de caracteres anexada al nombre de un archivo, usualmente precedida por un punto. Su principal función es diferenciar el contenido del archivo.
Facto	De común acuerdo, sin precedente jurídico.
FEC	Corrección de errores hacia adelante (en inglés, <i>forward error correction</i>), es un tipo de mecanismo de corrección de errores que permite su corrección en el receptor sin retransmisión de la información original.
FSK	Modulación por desplazamiento de frecuencia, en sus siglas en inglés, <i>frequency shift keying</i> .

GND	Tierra o común (en inglés, <i>ground</i>).
GPIB	Siglas en inglés de <i>general purpose instrumentation bus</i> (bus de instrumentación de propósito general), es un estándar para bus de datos digital de corto rango desarrollado para conectar dispositivos de testeo y medición.
Hardware	Conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático.
Hexadecimal	Sistema de numeración posicional que tiene como base el número 16. Vinculado actualmente con usos informáticos, telecomunicaciones y electrónica.
IDE	Ambiente de desarrollo integrado o entorno de desarrollo interactivo (en inglés, <i>integrated development environment</i>), aplicación informática que proporciona servicios integrales para el desarrollo de software.
IEEE	Siglas en inglés de Institute of Electrical and Electronics Engineers (Instituto de Ingenieros Eléctricos y Electrónicos), es una asociación mundial de técnicos e ingenieros dedicada a la estandarización y el desarrollo en áreas técnicas.

Interoperabilidad	Habilidad de dos o más sistemas o componentes para intercambiar información y utilizar la información intercambiada.
IOS	Siglas en inglés de iPhone Operating System, sistema operativo móvil desarrollado para teléfonos de la compañía Apple Inc.
IR	Siglas en inglés de <i>infrared</i> (infrarrojo), es un tipo de radiación electromagnética y térmica, de mayor longitud de onda que la luz visible, pero menor que la de las microondas.
ISM	Banda industrial, científico y médico (en inglés, <i>industrial, scientific and medical</i>), bandas reservadas internacionalmente para uso no comercial de radiofrecuencia electromagnética en áreas industrial, científica y médica.
LAN	Siglas en inglés de <i>local area network</i> (red de área local), es una red de computadoras que abarca un área reducida.
Linux	Sistema operativo basado en núcleo UNIX, cuyo código puede ser modificado y redistribuido libremente.

Memoria de programa	Destinada para almacenar el programa de usuario que puede contener datos alfanuméricos y textos variables, regularmente este tipo de memoria es volátil en fase de desarrollo, luego el programa finalizado es almacenado en memorias no volátiles.
Microchip Inc.	Empresa fabricante de microcontroladores, memorias y semiconductores analógicos.
MikroBasic PRO	Compilador de lenguaje BASIC con características para microcontroladores PIC de Microchip. Está diseñado para desarrollar, construir y depurar aplicaciones embebidas basadas en PIC.
Muestreo	Selección de una parte de una señal analógica en tiempos iguales que se consideran representativos de la señal original a la que pertenece, con la finalidad de procesarla, modificarla o analizarla.
NA	Normalmente abierto.
NC	Normalmente cerrado.
NPN	Transistor de unión bipolar de regiones tipo negativo, tipo positivo y tipo negativo.
Oscilador	Circuito activo que produce señales periódicas de voltaje o corriente.

Paridad	Suma de los bits 1 que tiene un número binario, si la suma es par, se dice que el número es paridad par, si la suma es impar se dice que el número es de paridad impar.
PCB	Circuito impreso (en inglés, <i>printed circuit board</i>).
PCB Wizard	Programa asistente para el diseño de circuitos impresos, a una o doble cara. Proporciona una amplia gama de herramientas que cubren todos los pasos tradicionales de producción de PCB, incluyendo dibujo esquemático, captura esquemática, colocación de componentes, enrutamiento automático, lista de materiales y la generación de informes de archivos para la fabricación.
PIC	Controlador de interfaz periférico (en inglés, <i>peripheral interface controller</i>), familia de microcontroladores tipo RISC fabricados por la empresa Microchip Technology Inc.
Pico red	Red de corto alcance formada por dos o más unidades o equipos que comparten un canal, es decir, que funcionan de forma síncrona y siguiendo la misma frecuencia de transmisión.

Potencia	Relación de paso de energía de un flujo por unidad de tiempo; es decir, la cantidad de energía entregada o absorbida por un elemento en un tiempo determinado.
Protocolo	Sistema de reglas que permiten que dos o más entidades dentro de un sistema de comunicación se comuniquen entre sí, para transmitir datos e información.
Puerto	Herramienta que permite manejar e intercambiar datos entre dispositivos y sus diferentes periféricos, o entre dos computadoras.
Red	Conjunto de equipos informáticos y programas conectados entre sí por medio de dispositivos físicos que envían y reciben datos e información.
Relé	Interruptor accionado por un electroimán.
Resonador	Dispositivo pasivo cuya impedancia cambia rápidamente con la frecuencia.
RFCOMM	Siglas en inglés de <i>radio frequency communication</i> (comunicación por radio frecuencia), es un protocolo de comunicación de transporte que proporciona sesenta conexiones simultáneas para dispositivos <i>bluetooth</i> .

RS-232	Siglas en inglés de <i>recommended standard 232</i> (estándar recomendado 232), es una interfaz que designa una norma para el intercambio de una serie de datos entre dos dispositivos.
SIG	Siglas en inglés de Special Interest Group, es una asociación privada sin ánimo de lucro que dirige el desarrollo de la tecnología inalámbrica <i>bluetooth</i> , además de implementar y comercializar la tecnología en sus productos.
Sintaxis	Grupo de normas que marcan las secuencias correctas de los elementos propios de un lenguaje de programación.
Software	Conjunto de programas y rutinas que permite a la computadora realizar determinadas tareas.
SPDT	Relé simple polo doble tiro.
SPP	Siglas en inglés de <i>serial port profile</i> (perfil de puerto serie), emula una línea serie y provee una interfaz de reemplazo de comunicaciones basadas en RS-232, con las señales de control típicas.
UART	Siglas en inglés de <i>universal asynchronous receiver-transmitter</i> (transmisor - emisor asíncrono universal), dispositivo que controla los puertos y dispositivos serie.

USB

Siglas en inglés de *universal serial bus* (modelo de aceptación de la tecnología), es un estándar industrial que define los cables, conectores y protocolos usados en un bus para conectar, comunicar y proveer de alimentación eléctrica entre computadoras, periféricos y dispositivos electrónicos.

Windows

Distribuciones de software para computadoras, servidores, teléfonos inteligentes; desarrollados por Microsoft. Sistema operativo desarrollador por Microsoft.

RESUMEN

Tomando como base las ventajas que ofrece el estándar de comunicación inalámbrica *bluetooth*, se diseña un dispositivo capaz de comunicarse inalámbricamente para transmitir o recibir información desde cualquier sistema de control dentro del campo industrial. En el capítulo uno se muestran los antecedentes, el concepto general de *bluetooth*, las normas por la cual se rige el estándar e información adicional que sirve de marco teórico para la elaboración del diseño base.

En el capítulo dos se presenta información específica referente al aspecto de comunicación del estándar *bluetooth*; el cual es serial, las velocidades de transmisión permisibles y otras características propias del *bluetooth*.

El planteamiento general del dispositivo a diseñar se observa en el capítulo tres, en el cual se encuentra las especificaciones del diseño como voltaje de operación, el módulo de comunicación *bluetooth* y generalidades del control de dispositivo. El capítulo tres es complementado por el cuatro, ya que, es en este donde se presenta el diseño de hardware, desde el control, luego las entradas y salidas digitales con las que cuenta el dispositivo, hasta el diseño final del dispositivo y circuito impreso correspondiente.

Por último, y no menos importante, el programa del microcontrolador, el cual se encarga de gestionar la información de las entradas y salidas, de establecer la comunicación con los maestros a partir del algoritmo base de ejecución.

OBJETIVOS

General

Diseñar un dispositivo *bluetooth* de comunicación bidireccional para el campo industrial.

Específicos

1. Reducir el costo de un dispositivo inalámbrico de comunicación destinado para el campo de la industria de la automatización y sistemas de control.
2. Crear un dispositivo portátil que sea compatible con diferentes sistemas de control maestro.
3. Tener la capacidad de transmisión bidireccional y opción a elegir el tipo de información a enviar o recibir por medio de software.

INTRODUCCIÓN

Con la finalidad de reducir costos, mano de obra y optimizar el proceso de comunicación al momento de realizar una conexión entre actuadores, sensores y periféricos de un sistema automatizado u otro sistema de control, se plantea el desarrollo del diseño de un dispositivo capaz de comunicarse bidireccionalmente e inalámbricamente en el campo industrial, con la capacidad de mejorar la velocidad de transmisión, lograr inmunidad al ruido y conseguir distancias mayores de comunicación.

Es por ello que se plantea el diseño de un dispositivo capaz de comunicarse inalámbricamente por medio de *bluetooth*, para transmitir o recibir información desde cualquier sistema de control dentro del campo industrial, obteniendo la ventaja de ser compatible con diferentes plataformas de trabajo, dotado de entradas y salidas digitales, y sobre todo la facilidad de interactuar con otros dispositivos.

1. MARCO TEÓRICO

1.1. Antecedentes

El origen del nombre de la tecnología *bluetooth* proviene de un vikingo de origen danés Haral Blatand (diente azul) quien en el siglo décimo unificó Dinamarca y Noruega. El nombre fue adoptado por la empresa Ericsson, que esperaba que el *bluetooth* unificara las telecomunicaciones y la industria de la computadora.

1.2. Concepto de *bluetooth*

Bluetooth es un estándar de facto concebido para establecer conexiones sin hilos de corto alcance entre un amplio número de dispositivos electrónicos, fijos y móviles. La tecnología *bluetooth* funciona a través de minúsculos y baratos transmisores y receptores de radio de alta frecuencia, baja potencia y corto alcance, implementados en un pequeño microchip de silicio que es integrado en teléfonos móviles, ordenadores personales y, en general, en todos los dispositivos electrónicos que se desee, directamente o a través de adaptadores.

La banda de radiofrecuencia utilizada es la de 2,45 GHz ISM (*industrial, scientific and medical*); en concreto las frecuencias entre 2 400 y 2 483,5 MHz en Estados Unidos y Europa (solo parte de esta banda está disponible en España y Francia), y entre 2,471 y 2,497 MHz en Japón. La banda de 2,45 GHz ISM está, por lo tanto, disponible prácticamente en todo el mundo sin necesidad de licencia. El ancho de banda de los canales dentro de la banda ISM es de

1 MHz, es decir, se soportan hasta 79 canales (23 en Japón) de radiofrecuencia; y se utiliza modulación binaria por desplazamiento en frecuencia o FSK (*frequency shift keying*).

La solución ha sido diseñada para ser totalmente operativa incluso en entornos muy ruidosos y sujetos a muchas interferencias electromagnéticas, utilizando el acceso múltiple por división en el código con salto en frecuencia o CDMA-FH (*code division multiple access–frequency hop*) y métodos de corrección de errores hacia delante o FEC (*forward error control*) muy avanzados. El sofisticado modo de transmisión adoptado por la especificación *bluetooth*, proporciona además, una alta seguridad de los datos transmitidos mediante encriptación y autenticación.

El estándar ha definido dos niveles de potencia; el nivel de potencia menor (1 mW) que cubre una distancia de alrededor de una habitación (hasta 10 metros) y el nivel de potencia mayor (100 mW) conseguido mediante un amplificador opcional que cubre una distancia de alrededor de una casa (hasta 100 metros). En ambos casos no está restringido a enlaces directos, es decir, funciona aunque hayan obstáculos intermedios.

El enlace es capaz de transmitir voz o datos (incluyendo imágenes) en tiempo real, con capacidad máxima por canal de alrededor de 720 Kbps o de 1 Mbps considerando los bits de control de errores, autenticación, entre otros.

Cualquier equipo electrónico o dispositivo con características *bluetooth* puede intercambiar información remota cuando esté al alcance de otro. Con solo encenderlo o encontrarse dentro del área de cobertura, el dispositivo busca a otros compatibles y establecen la comunicación, sincronizándose automáticamente en caso de que sea necesario.

La tecnología soporta varias conexiones simultáneas, tanto punto a punto como punto a multipunto. En la especificación actual, hasta siete dispositivos esclavos pueden comunicarse con un dispositivo maestro, formando lo que se denomina una picored. Dos o más dispositivos compartiendo el mismo canal constituyen una picored; siendo el maestro el encargado de controlar el tráfico, iniciar la comunicación y decidir qué esquema de salto de frecuencia utilizar. Por otro lado, pueden estar enlazadas de forma conjunta hasta diez de estas picoredes, formando redes dispersas que posibilitan comunicaciones entre configuraciones muy flexibles.

1.3. EI SIG

El fabricante de equipos de telecomunicaciones sueco Ericsson, ha sido el principal promotor de *bluetooth*, a raíz de un estudio iniciado en 1994 que pretendía conseguir una interfaz de radio de baja potencia y bajo coste que reemplazase el cable entre los teléfonos celulares y sus correspondientes accesorios. En febrero de 1998, cinco promotores, Ericsson, IBM, Intel, Nokia y Toshiba, forman un grupo de interés especial o SIG (*special interest group*), constituido por empleados voluntarios de las distintas compañías miembros.

Poco después, en mayo de 1998, *bluetooth* es lanzado como un estándar de licencia gratuita. Un año más tarde, en 1999, más de 650 compañías firmaron el acuerdo de subscriptores de esta tecnología, entre las que se encontraban las principales empresas de las distintas ramas de las industrias de la electrónica, informática y telecomunicaciones. Por otro lado, ese mismo año, los promotores se elevan a nueve, con la incorporación de 3Com, Lucent, Microsoft y Motorola. En la actualidad, más de 2 500 compañías han firmado el acuerdo de licencia, obteniendo el derecho a desarrollar y vender dispositivos *bluetooth*.

El principal objetivo del consorcio es establecer un estándar de facto para la interfaz aire y el software que lo controla, asegurando de esta forma la interoperabilidad entre los distintos dispositivos de los diferentes fabricantes. La versión 1.0 del estándar abierto *bluetooth* fue publicada en 1999.

La especificación *bluetooth* está disponible al público y sus regalías son gratuitas. La razón por la que *bluetooth* se ha establecido como un estándar abierto es que, en contraposición con los estándares propietarios, la aceptación por parte del mercado es mucho mayor. Para mantener la calidad y asegurar la compatibilidad, los productos *bluetooth* han de ser probados antes de que puedan ser comercializados con la marca *bluetooth*.

El convertirse en un miembro del SIG significa, básicamente, firmar un acuerdo de licencia sin coste que da derecho a utilizar la especificación *bluetooth* para desarrollar y fabricar productos y aplicaciones utilizando la especificación *bluetooth*. Por otro lado, también da derecho a influir en las especificaciones finales que adopten los promotores.

1.4. El mercado de *bluetooth*

Un estudio realizado por la compañía de investigaciones de mercado Frost & Sullivan estimó, en el año 2001 que el número de productos distribuidos a nivel mundial con tecnología *bluetooth* alcanzaría 1,2 millones de dispositivos, lo que se estimaba un volumen de negocio por 862 millones de dólares. Estos resultados eran de bastante significancia, pero aún por debajo de las previsiones iniciales de venta, se previó que el crecimiento acelerado del mercado se alcanzaría para el año 2003, que fue el año en el cual se solventaron los problemas de coste del silicio que tiene impacto directo en el precio de los dispositivos, los problemas de interoperabilidad y seguridad, al

igual que el límite en el tipo de aplicaciones. No obstante, se ha de considerar, que el proceso de desarrollo de *bluetooth* ha sido espectacularmente rápido puesto que no se estandarizó hasta 1998.

El más reciente estudio realizado sobre el mercado para los dispositivos *bluetooth* entre 2012 y 2018, revelan que en el 2012 se tenía un estimado de 3,5 billones de dispositivos que contaban con tecnología *bluetooth* y para el año 2018, según las estadísticas, se estima que el número de dispositivos que contaran con *bluetooth* ascenderá a 10 billones. Por otro lado, también ha de valorarse el hecho que Bluetooth es una tecnología orientada a la comunicación de voz y datos, es decir, está concebida para ofrecer conectividad a más dispositivos que los presentes en una LAN tradicional.

En la actualidad, el *bluetooth* ha alcanzado una alta aceptación por parte de los usuarios de los productos y son muchas las empresas implicadas en su desarrollo, con lo cual está asegurado el éxito a corto plazo para lo que se denomina la red de las cosas.

1.5. Las actuales aplicaciones del *bluetooth*

Las primeras aplicaciones de *bluetooth* están enfocadas a personas que viajan continuamente por negocios. Los dispositivos portátiles que integran el chip *bluetooth* posibilitan actualmente, entre otras, aplicaciones del tipo:

- Utilizar un teléfono tres en uno que funciona en casa como un inalámbrico con tarifa de teléfono fijo, como móvil con su correspondiente tarifa cuando se sale de casa y como *walkie-talkie* sin coste alguno cuando entra dentro del alcance de otro teléfono Bluetooth.

- Navegar por internet mediante computadoras personales fijas o portátiles sin conexión física de cables a una roseta de teléfono, facilitando de este modo la instalación y reduciendo la inversión necesaria en cableado.
- Sincronizar automáticamente y de forma totalmente transparente al usuario, un ordenador y un asistente digital personal; de forma que, por ejemplo, se transfieran los datos del día del asistente digital personal a la computadora personal y se encuentre actualizada antes de entrar en la oficina.
- Transferir documentos y presentaciones automáticamente a los participantes de una reunión sin necesidad de un cable.
- Navegar por internet a través de una red celular en una computadora portátil sin necesidad de cableado alguno entre computadora y teléfono móvil.
- Contestar las llamadas entrantes de un teléfono móvil con un auricular manos libres, desde el timón de un automóvil, sin necesidad de cables, integrada con la tecnología de reconocimiento de voz.
- Enviar o imprimir una postal instantáneamente utilizando una cámara digital.
- Utilizar el teléfono móvil como control remoto para operar los sistemas de iluminación, calefacción, alarmas, entre otros.
- Conectar dos o más computadoras portátiles o personales de casa o de una oficina sin necesidad de realizar ningún tendido de cable, para

transferir datos, permitir que varios usuarios jueguen simultáneamente en red, compartir recursos como impresoras o unidades de almacenamiento de datos.

En un futuro, cabe pensar también en aplicaciones más avanzadas, como acceder a determinadas zonas restringidas sin necesidad de introducir o enseñar la tarjeta de identificación, embarcar directamente en los vuelos sin necesidad de mostrar pasaporte, realizar pagos automáticamente mediante un monedero electrónico enviando la información de la tarjeta de crédito directamente al cajero o dependiente, automatizar máquinas industriales, entre otros.

1.6. Norma IEEE 802.11

La versión original del estándar IEEE (Instituto de Ingenieros Eléctricos y Electrónicos) 802.11 publicada en 1997 especifica dos velocidades de transmisión teóricas de 1 y 2 megabits por segundo (Mbps) que se transmiten por señales infrarrojas (IR). IR sigue siendo parte del estándar, si bien no hay implementaciones disponibles.

El estándar original también define el protocolo CSMA/CA (múltiple acceso por detección de portadora evitando colisiones) como método de acceso. Una parte importante de la velocidad de transmisión teórica se utiliza en las necesidades de codificación para mejorar la calidad de la transmisión bajo condiciones ambientales diversas, lo cual se tradujo en dificultades de interoperabilidad entre equipo de diferentes marcas. Estas y otras debilidades se corrigieron en el estándar 802.11b, que fue el primero de esta familia en alcanzar amplia aceptación entre los consumidores.

2. COMUNICACIÓN SERIAL

2.1. Definición

La comunicación serial es un protocolo muy común (no hay que confundirlo con el bus serial de comunicación, o USB) para comunicación entre dispositivos que se incluye de manera estándar en prácticamente cualquier computadora. La mayoría de las computadoras incluyen dos puertos seriales RS-232. La comunicación serial es también un protocolo común utilizado por varios dispositivos para instrumentación; existen varios dispositivos compatibles con GPIB que incluyen un puerto RS-232. Además, la comunicación serial puede ser utilizada para la adquisición de datos, si se usa en conjunto con un dispositivo remoto de muestreo.

El concepto de comunicación serial es sencillo. El puerto serial envía y recibe bytes de información un bit a la vez. Aún cuando esto es más lento que la comunicación en paralelo, que permite la transmisión de un byte completo por vez, este método de comunicación es más sencillo y puede alcanzar mayores distancias. Por ejemplo, la especificación IEEE 488 para la comunicación en paralelo determina que el largo de cable para el equipo no puede ser mayor a 20 metros, con no más de 2 metros entre cualquiera de los dos dispositivos; por otro lado, utilizando comunicación serial el largo del cable puede llegar a los 1 200 metros.

Típicamente, la comunicación serial se utiliza para transmitir datos en formato ASCII. Para realizar la comunicación se utilizan 3 líneas de transmisión: la primera para tierra (o línea de referencia), la segunda para transmitir y la

tercera para recibir. Debido a que la transmisión es asincrónica, es posible enviar datos por una línea mientras se reciben datos por otra. Existen otras líneas disponibles para realizar *handshaking*, o intercambio de pulsos de sincronización, pero no son requeridas. Las características más importantes de la comunicación serial son la velocidad de transmisión, los bits de datos, los bits de parada y la paridad. Para que dos puertos se puedan comunicar, es necesario que las características sean iguales.

2.1.1. Velocidad de transmisión (*baud rate*)

Indica el número de bits por segundo que se transfieren y se mide en baudios. Por ejemplo, 300 baudios representan 300 bits por segundo. Cuando se hace referencia a los ciclos de reloj, se está hablando de la velocidad de transmisión. Por ejemplo, si el protocolo hace una llamada a 4 800 ciclos de reloj, entonces el reloj está corriendo a 4 800 Hz, lo que significa que el puerto serial está muestreando las líneas de transmisión a 4 800 Hz. Las velocidades de transmisión más comunes para líneas de telefónicas son de 14 400, 28 800 y 33 600. Es posible tener velocidades más altas, pero se reduciría la distancia máxima posible entre los dispositivos. Las altas velocidades se utilizan cuando los dispositivos se encuentran uno junto al otro, como es el caso de dispositivos GPIB.

2.1.2. Bits de datos

Se refiere a la cantidad de bits en la transmisión. Cuando la computadora envía un paquete de información, el tamaño de ese paquete no necesariamente será de 8 bits. Las cantidades más comunes de bits por paquete son 5, 7 y 8 bits. El número de bits que se envían depende del tipo de información que se transfiere. Por ejemplo, el ASCII estándar tiene un rango de 0 a 127, es decir,

utiliza 7 bits; para ASCII extendido es de 0 a 255, lo que utiliza 8 bits. Si el tipo de datos que se está transfiriendo es texto simple (ASCII estándar), entonces es suficiente utilizar 7 bits por paquete para la comunicación. Un paquete se refiere a una transferencia de byte, incluyendo los bits de inicio/parada. Debido a que el número actual de bits dependen del protocolo que se seleccione, el término paquete se usa para referirse a todos los casos.

2.1.3. Bits de parada

Usado para indicar el fin de la comunicación de un solo paquete. Los valores típicos son 1, 1,5 o 2 bits. Debido a la manera como se transfiere la información a través de las líneas de comunicación y que cada dispositivo tiene su propio reloj, es posible que los dos dispositivos no estén sincronizados. Por lo tanto, los bits de parada no solo indican el fin de la transmisión, sino además dan un margen de tolerancia para esa diferencia de los relojes. Mientras más bits de parada se usen, mayor será la tolerancia a la sincronía de los relojes, sin embargo la transmisión será más lenta.

2.1.4. Paridad

Es una forma sencilla de verificar si hay errores en la transmisión serial. Existen cuatro tipos de paridad: par, impar, marcada y espaciada. La opción de no usar paridad alguna también está disponible. Para paridad par e impar, el puerto serial fijará el bit de paridad (el último bit después de los bits de datos) a un valor para asegurarse que la transmisión tenga un número par o impar de bits en estado alto lógico. Por ejemplo, si la información a transmitir es 011 y la paridad es par, el bit de paridad sería 0 para mantener el número de bits en estado alto lógico como par. Si la paridad seleccionada fuera impar, entonces el bit de paridad sería 1, para tener 3 bits en estado alto lógico.

La paridad marcada y espaciada en realidad no verifican el estado de los bits de datos; simplemente fija el bit de paridad en estado lógico alto para la marcada y en estado lógico bajo para la espaciada. Esto permite al dispositivo receptor conocer de antemano el estado de un bit, lo que serviría para determinar si hay ruido que esté afectando de manera negativa la transmisión de los datos o si los relojes de los dispositivos no están sincronizados.

2.2. La comunicación serial en tecnología *bluetooth*

Actualmente, el estándar *bluetooth* define un cierto número de perfiles de aplicación (denominados perfiles *bluetooth*) para definir qué tipos de servicios ofrece un dispositivo bluetooth. Por lo tanto, cada dispositivo puede admitir múltiples perfiles, entre estos perfiles se encuentra el SPP o perfil de puerto serial.

El perfil de puerto serial define los requisitos necesarios en dispositivos *bluetooth* para la conexión mediante cable serie RFCOMM entre dos dispositivos pares. Los requisitos se expresan en función de los servicios prestados a las solicitudes, y mediante la definición de las características y procedimientos que se requieran para la interoperabilidad entre dispositivos *bluetooth*. En dicho perfil se emula una línea serie y se provee una interfaz de reemplazo de comunicación basadas en RS-232, con señales de control típicas.

También, el perfil SPP mantiene un reemplazo inalámbrico para la existencia del puerto serie RS-232 basado en la aplicación de comunicación serial y en el control de señales. SPP proporciona la base para el DUN (perfil de gestión de redes por vía telefónica), FAX, HSP (perfil auricular). El perfil SPP soporta una velocidad de transmisión de datos de hasta 128 Kbit/s y es dependiente del perfil genérico GAP.

2.2.1. Protocolo RFCOMM

EL RFCOMM provee una emulación del puerto serial sobre el protocolo L2CAP. El protocolo está basado en el estándar TS 07.10. RFCOMM es un simple protocolo de transporte, con la ventaja de que provisiona la emulación de 9 circuitos de comunicación del RS-232 (EIA/TIA-232E) de puertos seriales.

El protocolo soporta más de 60 conexiones simultáneas entre dos dispositivos *bluetooth*. El número de conexiones que puede ser usada simultáneamente en un dispositivo *bluetooth* es implementación específica.

2.2.1.1. Circuitos que emula el RFCOMM

RFCOMM emula los 9 circuitos de comunicación de una interfaz RS-232. Los circuitos se enlistan a continuación.

Tabla I. Circuitos emulados con RFCOMM

PIN	NOMBRE DEL PIN DEL CIRCUITO
102	<i>Signal common</i>
103	<i>Transmit data (TD)</i>
104	<i>Received data (RD)</i>
105	<i>Request to send (RTS)</i>
106	<i>Clear to send (CTS)</i>
107	<i>Data set ready (DSR)</i>
108	<i>Data terminal ready (DTR)</i>
109	<i>Data carrier detect (CD)</i>
125	<i>Ring indicator (RI)</i>

Fuente: elaboración propia.

3. PLANTEAMIENTO DEL DISPOSITIVO

3.1. Generalidades del diseño

El dispositivo dispondrá de una fuente regulable de 5 VDC para la alimentación de todos sus componentes, se utilizará un PIC 16F690 para el control del mismo y un módulo *bluetooth* HC-05 para el establecimiento de la comunicación inalámbrica. Podrá enviar y recibir datos digitales de hasta 4 bits simultáneamente, a una velocidad de transmisión de 9 600 baudios por segundo y una distancia de transmisión máxima de 30 metros.

3.1.1. Especificaciones del dispositivo

Se describe a continuación cada una de las especificaciones con las que contará el dispositivo a diseñar.

3.1.1.1. Voltaje de operación

Considerando que un dispositivo para uso industrial pero que contará con muchos componentes con voltaje nominal para dispositivos electrónicos, el voltaje de operación para el módulo será de 5 VDC.

Tendrá dos entradas para voltaje, una entrada de alimentación exclusivamente para voltaje de 5 VDC y otra con fuente regulada a 5 VDC con voltaje de entrada de 8 a 12 VDC y una corriente máxima suministrada de 1,5 amperios, dicho regulador viene en empaquetado TO-220.

3.1.1.1.1. Entrada exclusiva 5 VDC

El dispositivo tendrá borneros exclusivos para voltaje de 5 VDC, esto para que, en caso de daño en la fuente regulada DC/DC, pueda utilizarse como una alternativa más para la alimentación del módulo.

3.1.1.1.2. Entrada de 8 a 12 VDC

La entrada de 8 a 12 VDC estará provista con un regulador LM7805, que es un regulador de DC/DC de hasta 1,5 amperios y un voltaje de salida de 4,8 a 5,2 VDC.

3.1.1.2. Módulo de comunicación *bluetooth*, HC-06

Para el establecimiento de la comunicación *bluetooth*, se utilizará un módulo serial con una tasa de comunicación de 9 600 baudios, la alimentación del módulo es de 5 VDC, utiliza un chip CSR. El módulo es fabricado y distribuido alrededor del mundo por Wavesen y tiene un costo inferior a \$ 8,00.

Figura 1. Módulo HC-06

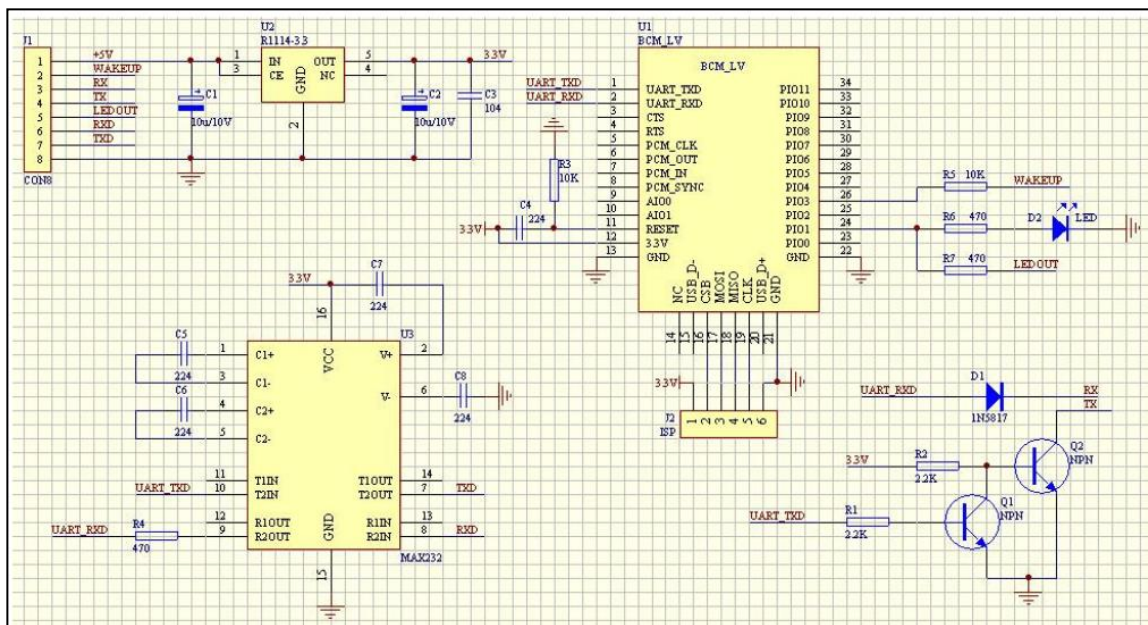


Fuente: ElectroniLab. *Módulo hc06*. <http://electronilab.co/tienda/modulo-bluetooth-hc-06-serial-rs232ttl/>. Consulta: 12 de abril de 2014.

La corriente de operación es de 40 mA, lo que ofrece un bajo consumo de energía, la cual es una ventaja.

A continuación se presenta el diagrama electrónico del dispositivo.

Figura 2. Diagrama de conexión del HC-06



Fuente: Guangzhou HC Information Technology. *Diagrama hc06*.

<https://www.olimex.com/Products/Components/RF/BLUETOOTH-SERIAL-HC-06/resources/hc06.pdf>. Consulta: 12 de abril de 2014.

Dicho módulo cuenta con los siguientes pines:

Tabla II. **Pines del HC-06**

PIN	NOMBRE
1	KEY
2	VCC
3	GND
4	TXD
5	RXD
6	STATE

Fuente: elaboración propia.

3.1.1.2.1. Pin KEY

La función del pin KEY es eliminar la última validación de conexión establecida; es decir, si ya se ha conectado un dispositivo al módulo y se vuelve a conectar, este recuerda la validación sin necesidad de solicitar el ingreso de clave. Pero si el pin KEY es conectado a la alimentación, dicha validación se borrará y al intentar conectar de nuevo el dispositivo, el módulo solicitará de nuevo la contraseña para emparejarlo.

3.1.1.2.2. Pin VCC

Entrada de alimentación positiva para el caso del HC-06 tiene una alimentación de 5 VDC, teniendo claro que el chip de control de *bluetooth* funciona con un voltaje de 3,3 VDC. Por ello, el módulo cuenta con un regulador R1114-3,3 que baja de 5 VDC a una salida a 3,3 VDC con una corriente máxima de 150 mA.

3.1.1.2.3. Pin GND

Es el pin de entrada de alimentación negativa, por lo que debe al conectarlo debe ir enchufado siempre al negativo de toda fuente de alimentación de 5 VDC.

3.1.1.2.4. Pin TXD

Pin de recepción de datos transmitidos vía UART por el microcontrolador, es importante no confundirlo como la transmisión de datos del módulo hacia el microcontrolador.

3.1.1.2.5. Pin RXD

Es el pin de transmisión de datos hacia el microcontrolador, es decir, el pin del cual el microcontrolador recibe la información. Es un detalle bastante importante que es una información que no está detallado en la hoja técnica del dispositivo módulo y que si no se presta atención, no puede funcionar adecuadamente la transmisión al momento de conectarlo.

3.1.1.2.6. Pin STATE

Este último pin es utilizado para conocer el estado del módulo, ya que si el dispositivo se encuentra desconectado, el pin estado se encontrará a 0 VDC, mientras que si se encuentra conectado estará en estado 5 VDC.

3.1.1.3. Generalidades del control del dispositivo

El control del dispositivo es por medio del PIC 16F690, que es un microcontrolador de 20 pines, de 8 bits de instrucciones tipo CMOS con tecnología *nanoWatt*.

3.1.1.3.1. Velocidad de operación

El microcontrolador puede operar a velocidades de 32 kHz hasta 20 MHz. Para rangos de 32 kHz a 8 Mhz puede ser seleccionado por medio de software y no necesita de oscilador externo. Para el caso de velocidades por arriba de 8 MHz a 20 MHz es necesario utilizar oscilador externo.

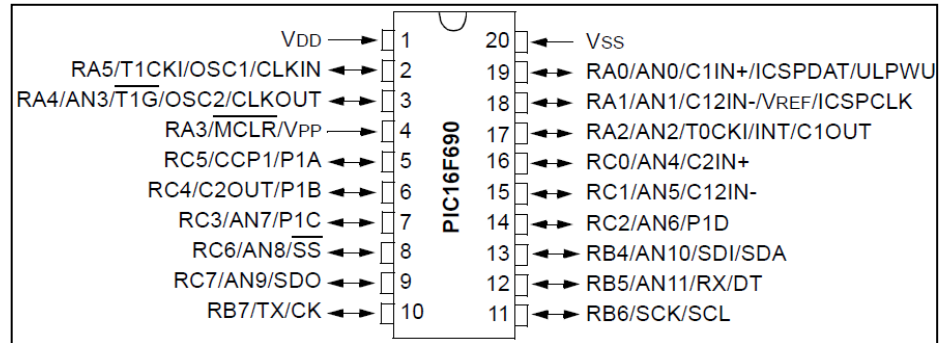
Para el diseño que se plantea, es necesaria una velocidad de 20 MHz con oscilador externo.

3.1.1.3.2. Entradas/salidas

Cuenta con 17 pines que tienen doble propósito entradas/salidas y un pin que funciona únicamente como entrada. Tres puertos A, B y C; el puerto A posee 6 bits, el puerto B es de 4 bits y el puerto C cuenta con 8 bits.

A continuación se presenta el diagrama de pines correspondiente al microcontralor.

Figura 3. Diagrama de pines del PIC16F690



Fuente: MicroChip. Hoja de datos PIC16f690.

<http://ww1.microchip.com/downloads/en/DeviceDoc/41262C.pdf>. Consulta: 15 de mayo de 2015.

3.1.1.3.3. Otras especificaciones del microcontrolador

A continuación se presentan otras especificaciones importantes del microcontrolador.

Tabla III. Especificaciones de entradas, salidas y memoria

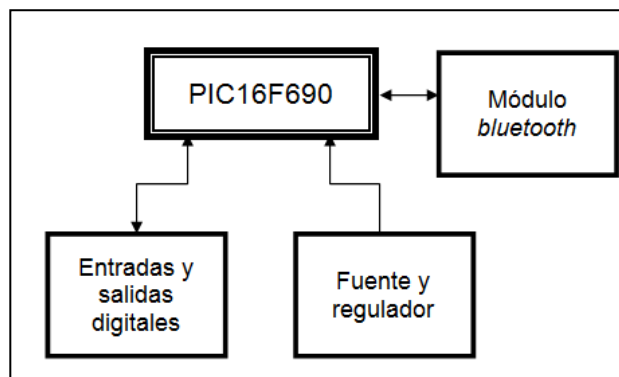
	Memoria de programa	Memoria de datos		I/O	10-bit A/D (ch)	Comparadores
	Flash (palabras)	SRAM (bytes)	EEPROM (bytes)			
PIC 16F690	4096	256	256	18	12	2

Fuente: elaboración propia.

3.2. Diagrama de bloques del dispositivo

Debido a que será un dispositivo multipropósito, se contará con varias entradas y salidas digitales. También, como pretende ser un dispositivo portátil y pequeño, la fuente que proveerá de energía a la circuitería constará de batería y regulador de voltaje, convenientemente se puede modificar para que el dispositivo tenga un cargador.

Figura 4. Diagrama de bloques del dispositivo



Fuente: elaboración propia.

4. DISEÑO DE HARDWARE

4.1. Control del dispositivo

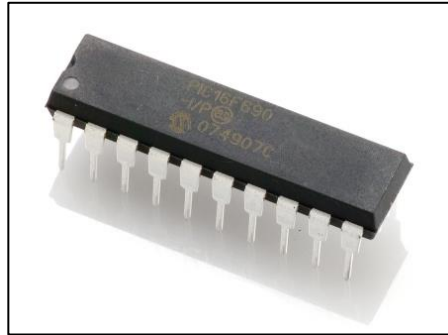
En el capítulo anterior se presentaron las generalidades de dispositivo. A continuación se presentará con detalle el diseño de hardware para el dispositivo.

4.1.1. El PIC16F690 como control de dispositivo

Se ha elegido el PIC16F690 como el controlador del dispositivo, ya que es un microcontrolador de la gama CMOS, de bajo costo y bajo consumo de energía porque utiliza la tecnología *nanoWatt*. Además, por ser un controlador de 8 bits ofrece un gran desempeño y la versatilidad de programación y aplicación.

El PIC16F690 es un microcontrolador de marca Microchip de 20 pines, basado en memoria *flash* con versión en empaquetado DIP (*dual in-line package*), el cual es común en circuitos integrados.

Figura 5. **DIP20 PIC16F690**



Fuente: Nerokas. <http://store.nerokas.co.ke/image/cache/data/pic16f690-600x600.jpg>.

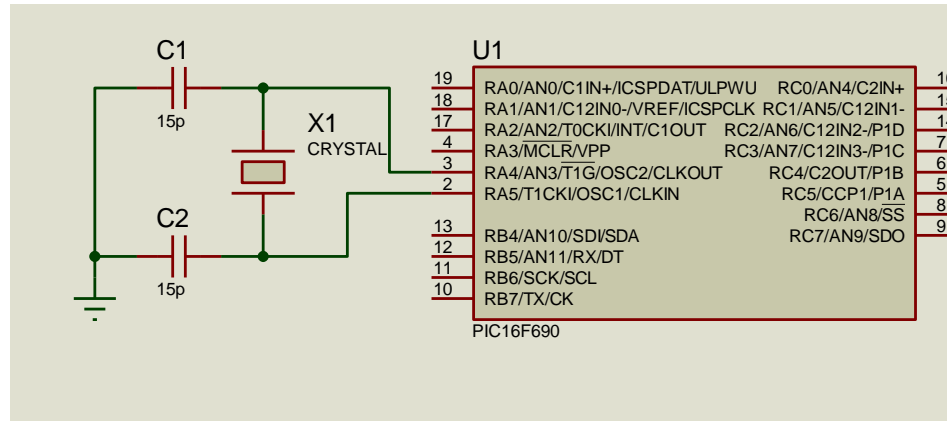
Consulta: 22 de julio de 2014.

4.1.2. Reloj del microcontrolador

Para la velocidad de operación del dispositivo y configurar la velocidad de transmisión de 9 600 bits por segundo, se debe utilizar un reloj de 20 MHz en configuración HS (*high-speed cristal/resonator on RA4/OSC2/CLKOUT and RA5/OSC1/CLKIN*), ya que a velocidades superiores a 8 MHz es necesario un cristal externo, puesto que únicamente en el rango de velocidades de 32 KHz a 8 MHz puede seleccionarse por software el oscilador interno de precisión.

A continuación se presenta la conexión del hardware del reloj externo necesario para la operación del dispositivo:

Figura 6. **Conexión de oscilador externo 1**



Fuente: elaboración propia, empleando ISIS Profesional.

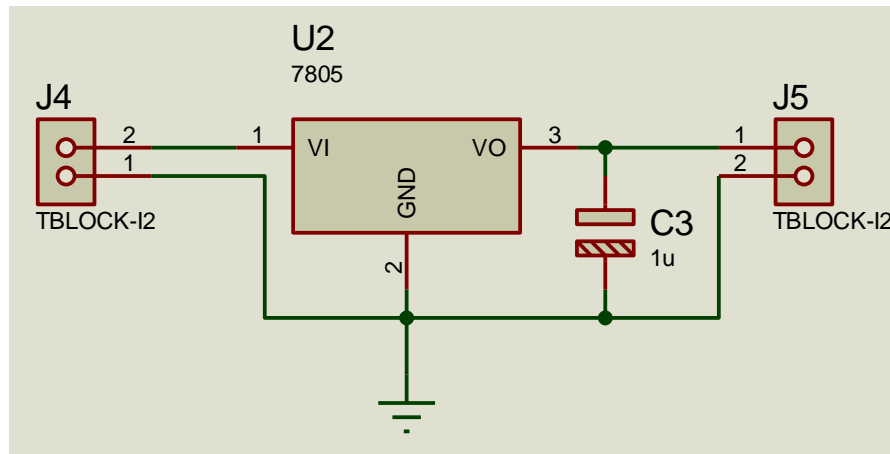
Como se observa en la figura 6, el cristal debe ir conectado entre RA4 y RA5 del microcontrolador, también es necesario conectar dos capacitores de 15 pF de cada pata del cristal hacia negativo, ya que para la operación del cristal de cuarzo de 20 MHz, el fabricante especifica utilizar capacitores en el rango de 15 a 33 pF para la operación correcta.

Cabe resaltar que para la máxima velocidad de operación del microcontrolador 16F690, el consumo de energía es el máximo para dicho dispositivo que llega a ser de 6 mA, es decir unos 30 mW.

4.2. Fuente de alimentación del dispositivo

El dispositivo estará provisto de una fuente regulada de 5 VDC, a la cual a su entrada se puede colocar voltaje de 8 a 12 VDC. Dicho regulador es el LM7805 el cual provee una salida máxima de 1,5 amperios. Para la correcta regulación del dispositivo se recomienda utilizar a la salida del voltaje regulado de un valor de 1 μ F.

Figura 7. **Conexión de oscilador externo 2**



Fuente: elaboración propia, empleando ISIS Profesional.

4.3. Entradas digitales

El dispositivo posee cuatro entradas digitales con resistencia de *pull-up* a negativo para garantizar los estados digitales a los puertos del microcontrolador, dichas resistencias son de un valor de 330 ohm $\frac{1}{4}$ de watt, las entradas soportan exclusivamente voltaje de 5 VDC las cuales son no negadas.

Cada una de las entradas digitales está conectada a un borne para la realización de conexiones rápidas en el dispositivo. Dichas entradas están a su vez conectadas con un pin del puerto C del microcontrolador de la siguiente manera.

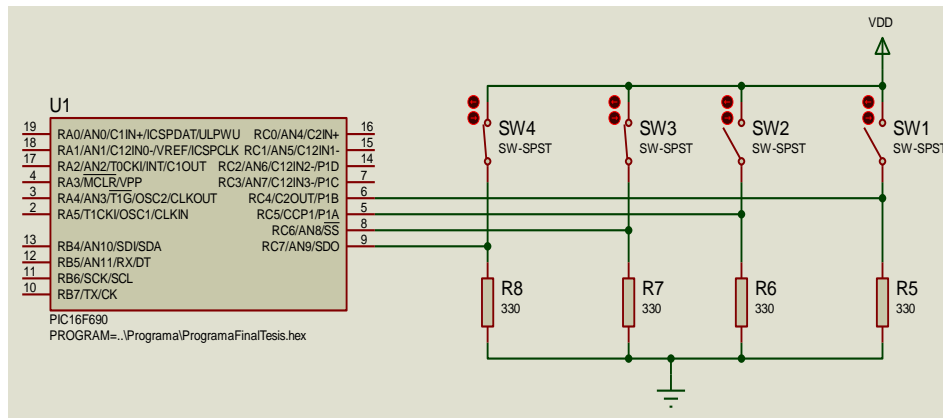
Tabla IV. Especificaciones de entradas, salidas y memoria

NÚMERO DE ENTRADA	PIN EN MICROCONTROLADOR
1	RC4
2	RC5
3	RC6
4	RC7

Fuente: elaboración propia.

A continuación se presenta el circuito relativo a las entradas digitales del dispositivo.

Figura 8. Entradas digitales



Fuente: elaboración propia, empleando ISIS Profesional.

4.4. Salidas digitales

Puesto que los pines de puerto en configuración de salida del microcontrolador soportan una corriente máxima de 20 mA, se han configurado las salidas digitales a modo de optimizar este consumo y ser capaz de accionar el relevador de 5 VDC. Al realizar el cálculo de corriente de base del transistor NPN 2N3904 que accionará la bobina del relé, se obtiene el siguiente resultado.

$$I_B = \frac{5 - V_{BE}}{330} = \frac{5 - 0,7}{330} = 13,03 \text{ mA}$$

Como se observa, la corriente consumida por la salida del microcontrolador es un poco más de la mitad del máximo permisible.

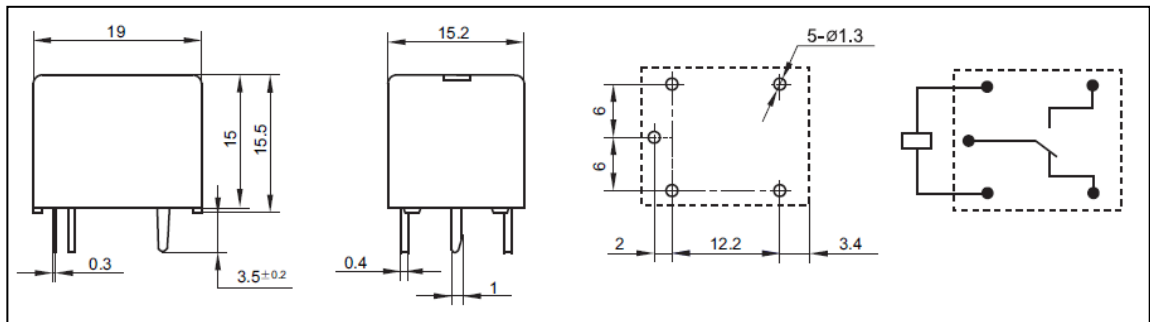
Las salidas digitales cuentan con una luz piloto que acusa de que el relevador correspondiente a la salida está accionado. Los relevadores utilizados para el diseño son SPDT (simple polo doble tiro), es decir que cuentan con un pin común, uno normalmente abierto (NA) y uno normalmente cerrado (NC), además de su bobina de excitación.

4.4.1. Relevadores de salida

Los relevadores seleccionados para el dispositivo son los RH-005C tipo SPDT, que cuentan con una bobina de 5 VDC de 70 ohm \pm 10 %, lo que es una ventaja para el consumo de corriente.

Su diseño compacto ofrece la posibilidad de integrarlos al diseño, ya que no ocupan mucho espacio dentro del PCB (*printed circuit board*).

Figura 9. **Medidas físicas del relevador RH-005C**



Fuente: Hongfa Relay. http://www.hongfa.com/pro/pdf/HF3FF_en.pdf. Consulta: 02 agosto de 2014.

En cuanto a la capacidad de intensidad de sus contactos, se observan las siguientes características importantes:

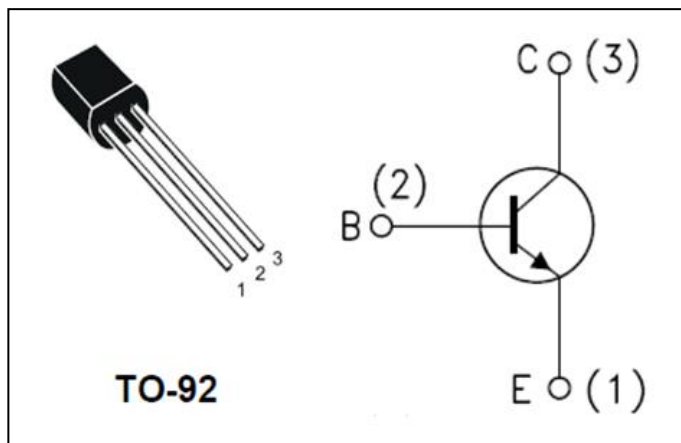
- La resistencia de los contactos es de 100 mΩ máximo (medición especificada por el fabricante a 1 A a 6 VDC).
- Intensidad máxima en los contactos es de 10 A/28 VDC y 10 A/125 VAC.
- El relé es capaz de operar hasta temperaturas de 70 grados Celsius en humedad relativa desde 5 al 85 %.

Al observar las características, se evidencia la capacidad de operación para ambientes industriales, a lo que está orientado el dispositivo diseñado.

4.4.1.1. Circuito de excitación de bobinas de relevadores

Como se ha mencionado anteriormente, para accionar las bobinas de los relés de salidas digitales se han seleccionado los transistores 2N3904, los cuales son transistores de uso general con empaquetado TO-92 del tipo NPN.

Figura 10. Transistor 2N3904 NPN



Fuente: Spark Fun. <https://www.sparkfun.com/datasheets/Components/2N3904.pdf>.

Consulta: 10 de agosto de 2014.

A continuación la nomenclatura de los pines del transistor.

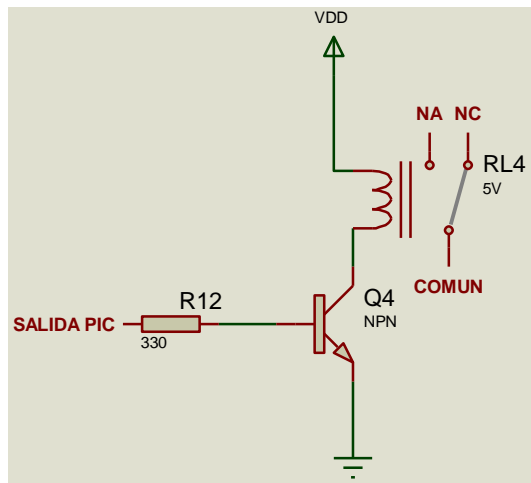
Tabla V. **Nomenclatura de pines del transistor**

NÚMERO DE PIN	NOMBRE DE PIN
(1) E	Emisor
(2) B	Base
(3) C	Colector

Fuente: elaboración propia.

La configuración para excitación de los relés de salida digitales es de emisor común para los transistores, ya que es la polarización más sencilla de las existentes para el transistor.

Figura 11. **Circuito de salida digital por relé**



Fuente: elaboración propia, empleando ISIS Profesional.

Como se observa en la figura 11, para accionar el relé de la salida es necesaria una entrada en la resistencia de 330 ohm de 5 VDC que habilite la alimentación de la bobina. Para ello se utiliza la salida del PIC16F690 y para el

caso del dispositivo son necesarias 4 salidas digitales de 5 VDC del microcontrolador.

En la tabla VI se presentan las salidas digitales que se habilitarán.

Tabla VI. **Pines habilitadores de salidas digitales**

PIN SALIDA DEL PIC	SALIDA DIGITAL POR RELE
RC0	OUT 0
RC1	OUT 1
RC2	OUT 2
RC3	OUT 3

Fuente: elaboración propia.

Por lo tanto, el dispositivo contará con 4 salidas digitales por relé accionadas cada una por un pin configurado como salida del microcontrolador.

4.5. Listado de componentes del dispositivo

Se presenta ahora el listado de componentes para realizar el circuito impreso.

Tabla VII. **Listado de componentes del dispositivo**

No.	DESCRIPCIÓN	CANTIDAD
1	Relé de 5 VDC RH-005C	4
2	Transistor 2N3904	4
3	Resistencia 330 Ω 1/4 W	12
4	Regulador 7805CT	1
5	Socket ZIF 20 pines 3M	1
6	PIC16F690	1
7	Led 3 mm	4
8	Cristal de 20 MHz	1
9	Modulo Bluetooth HC-06	1
10	Capacitor de 1 μ F 16 VDC	1
11	Capacitor cerámico 15 pF	2
12	Terminal tipo bloque	20

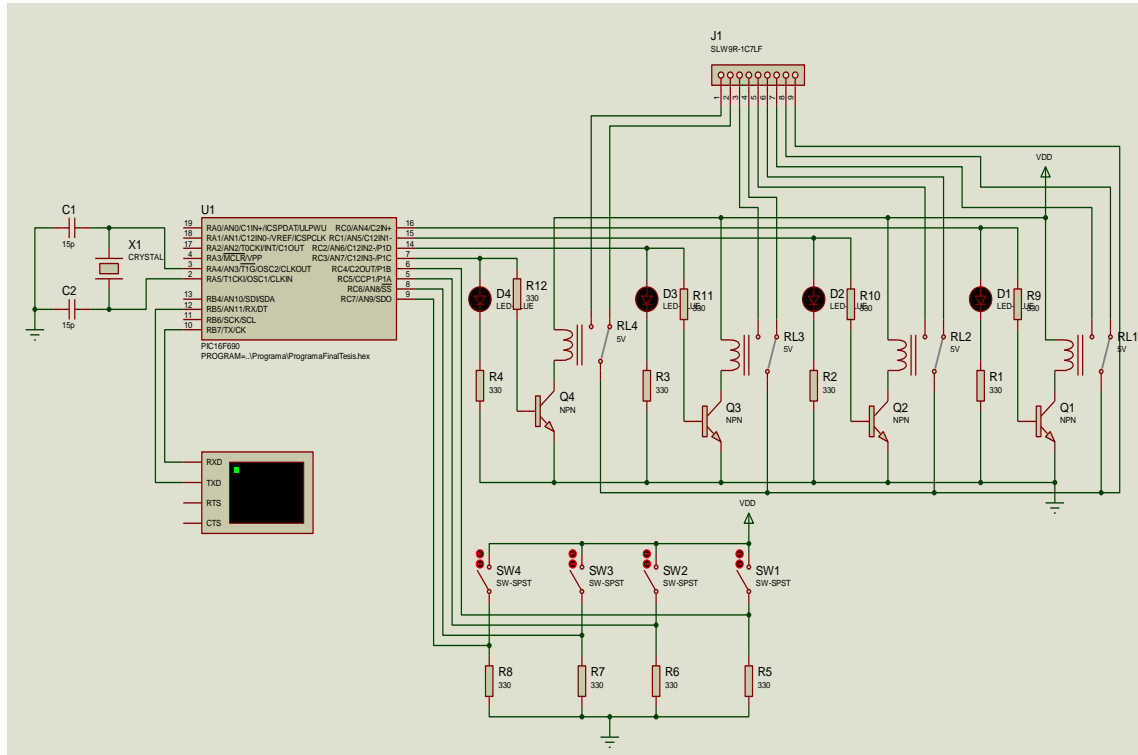
Fuente: elaboración propia.

4.6. Circuito final del dispositivo

Se ha completado el hardware de cada segmento del dispositivo, ahora es momento de concatenarlos para darle forma final al dispositivo y tener el diseño completado.

Por ello, a continuación se presenta el circuito final del dispositivo planteado en el presente trabajo, para luego proceder con el diseño del circuito impreso.

Figura 12. Circuito final del dispositivo

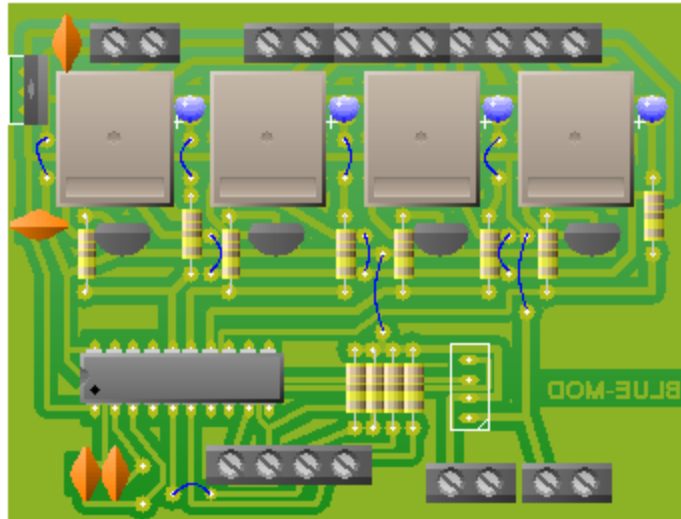


Fuente: elaboración propia, empleando ISIS Profesional.

4.7. Diseño de circuito impreso del dispositivo

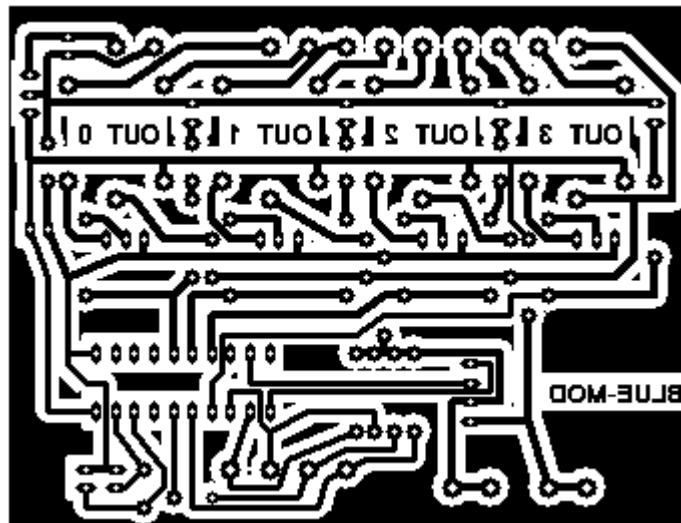
Para el diseño del circuito se utilizó el programa PCB Wizard y se realizó de forma manual para optimizar el espacio de la placa, para obtener un diseño compacto y muy bien estructurado. Se observa que las entradas digitales se encuentran en la parte baja del circuito impreso y las salidas digitales en la parte alta, tal y como se encuentra en la mayoría de dispositivos industriales.

Figura 13. Circuito impreso, vista real



Fuente: elaboración propia, PCB Wizard.

Figura 14. Circuito impreso, vista de pistas área de soldadura



Fuente: elaboración propia, empleando PCB Wizard.

5. PROGRAMA CONTROLADOR DEL DISPOSITIVO

En este último capítulo se presenta el software desarrollado para el control del dispositivo, para ello se ha utilizado el compilador MikroBasic PRO para PIC en su versión 6.4.0.

5.1. Algoritmo del programa de control

El algoritmo ha sido planteado de forma tal que optimice el proceso y el programa a implementar para el control del dispositivo, ya que un programa optimizado representa más rapidez del procesamiento en el microcontrolador y menos consumo de memoria interna del mismo.

A continuación se enlistan las actividades en forma secuencial del algoritmo planteado.

- Inicio
- Declaración de variables
- Etiqueta de programa principal
- Verificación de entrada de dato en puerto serial
- Lectura de dato en puerto
- Accionamiento de bandera de confirmación de lectura realizada
- Verificación de bandera accionada de lectura realizada
- Acción a realizar en función al dato leído
- Desactivación de bandera de lectura realizada
- Envío de datos a puerto serial

5.1.1. Inicio

Nombre del programa e inicialización y declaración del mismo.

5.1.2. Declaración de variables

En esta parte del algoritmo se deben declarar las variables necesarias para el funcionamiento del programa para el caso del dispositivo son necesarias tres.

- Variable de entrada de dato serial
- Variable de salida de dato serial
- Variable de confirmación de lectura de dato serial

Las tres variables propuestas deben ser del tipo byte, ya que el protocolo serial envía y recibe un byte a la vez.

5.1.3. Etiquetas de programa principal

Esta sirve para identificar el punto donde se encuentra el programa a ser ejecutado. A continuación se ejecutan por única vez las configuraciones de entradas y salidas de los puertos del microcontrolador, al igual que la inicialización del puerto UART a la velocidad de transmisión de 9 600 baudios.

5.1.4. Verificación de entrada de dato en puerto serial

Es el punto de decisión en el cual si existe un dato en el puerto por leer entra en ejecución de la respectiva lectura.

5.1.5. Lectura de datos en puerto

En este proceso se le asigna el dato en puerto a la variable de entrada de dato serial, para luego verificar la función a realizar en las salidas digitales.

5.1.6. Accionamiento de bandera de confirmación de lectura realizada

Una vez realizada la lectura del dato; se acciona la bandera de confirmación de lectura realizada, la cual indica que efectivamente se realizó y se guardó el dato en la variable correspondiente.

5.1.7. Verificación de bandera accionada de lectura realizada

En esta sección del algoritmo se verifica si la bandera de lectura realizada está o no activada, en caso positivo se realiza el paso siguiente. En caso contrario, el algoritmo conduce a la sección de envío de datos al puerto serial.

5.1.8. Actividad a realizar en función al dato leído

Una vez verificada la bandera que el dato en la entrada del puerto serial fue leído, se procede a realizar la actividad en función del dato entrante, es decir que para cierto dato el microcontrolador realizara una actividad distinta.

5.1.8.1. Actividades según dato entrante

Los datos entrantes serán caracteres, es decir un dato de 8 bits (1 byte), por lo cual la tabla de datos esperados relacionados a una actividad específica queda en función de la tabla VIII.

Tabla VIII. **Actividad a realizar el microcontrolador en función al dato de entrada esperando en el puerto serial**

DATO DE ENTRADA	ACTIVIDAD
"0"	RCO activado
"1"	RCO desactivado
"2"	RC1 activado
"3"	RC1 desactivado
"4"	RC2 activado
"5"	RC2 desactivado
"6"	RC3 activado
"7"	RC3 desactivado

Fuente: elaboración propia.

Una vez realizada la acción correspondiente, se procede a realizar la actividad siguiente del algoritmo.

5.1.9. Desactivación de bandera de lectura realizada

Luego de haber realizado la acción correspondiente según el dato de entrada del puerto serial, se procede a desactivar la bandera de lectura realizada. Esto con el fin de que la acción sea realizada una vez por dato entrante y así optimizar el programa.

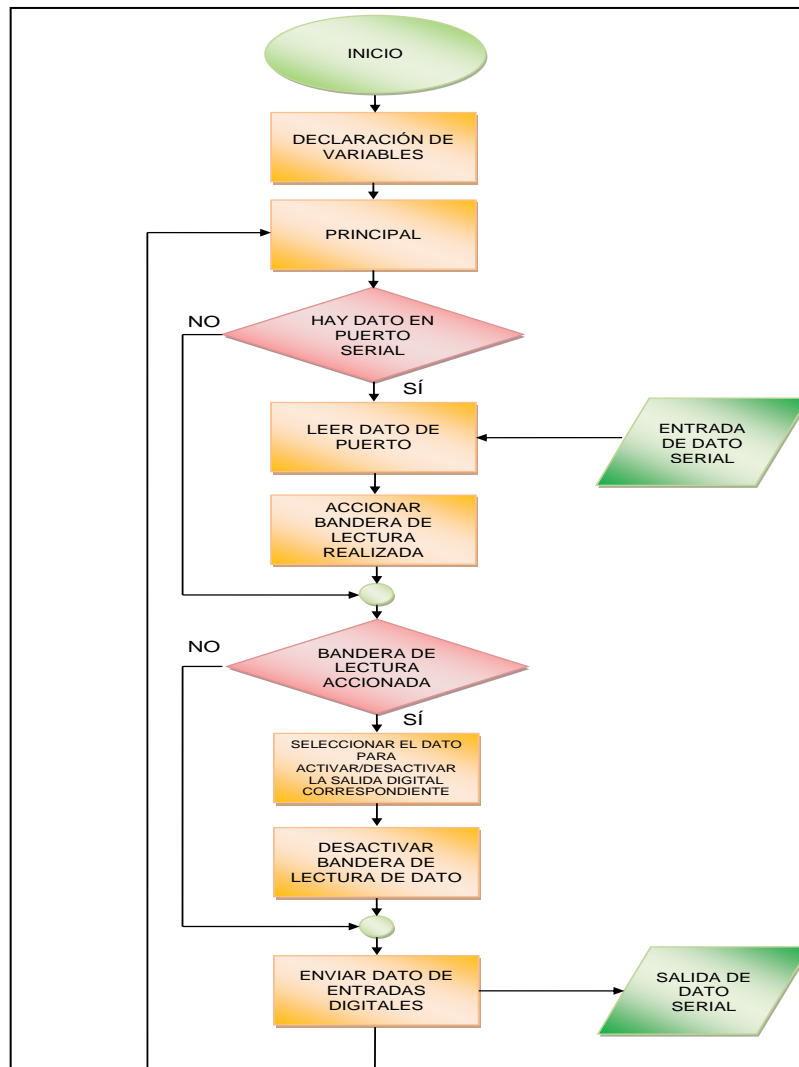
5.1.10. Enviar dato de entradas digitales

Por último, se envía al puerto serial del microcontrolador el dato de lectura de las entradas digitales correspondientes al módulo. Estos son los bits 4, 5, 6 y 7 del puerto c del microcontrolador, dichas entradas representan estados lógicos de periféricos digitales que puedan conectarse al módulo.

5.1.11. Representación gráfica del algoritmo

A continuación se presenta el algoritmo del programa de forma gráfica, ya que se trata de un programa secuencial, el modo de ejecución del mismo es de arriba hacia abajo, en forma ordenada.

Figura 15. Algoritmo del programa de control



Fuente: elaboración propia.

5.2. Programa de control desarrollado en MikroBasic Pro para PIC

Es necesario, una vez planteado el algoritmo que se utilizará de base para el desarrollo del programa de control para el microcontrolador, escoger el IDE o compilador a usar. Para este caso en el cual se diseña un módulo de comunicación *bluetooth* con enlace por puerto serial, se ha decidido implementarlo con el compilador MikroBasic.

5.2.1. Aspectos importantes del compilador

MikroBasic Pro es un compilador basado en lenguaje de alto nivel Basic que provee todos los aspectos necesarios para microcontroladores PIC de la marca Microchip, fue diseñado con el fin de desarrollar, implementar y optimizar aplicaciones embebidas al PIC.

El entorno de desarrollo del compilador ofrece ciertas características importantes como:

- Sintaxis fácil de aprender
- Programa de desarrollo amigable y de fácil manejo
- Código óptimo y eficiente
- Muchos equipos y bibliotecas compatibles con el IDE
- Documentación completa para implementación de aplicaciones

Por estos y otros aspectos relevantes se escoge a MikroBasic como el compilador para el desarrollo del programa de control del dispositivo *bluetooth*.

5.2.2. Código del programa desarrollado para el dispositivo *bluetooth*

A continuación se presenta el código correspondiente al programa implementado en el microcontrolador para control del dispositivo. Se adjuntan comentarios que indican qué realiza cada línea de código.

Figura 16. Código programa de control desarrollado en MikroBasic

```
1: program ProgramaFinalTesis
2:
3: dim DatoEnviar as byte           'Variable de dato enviar a puerto serial
4: dim DatoEntrada as byte         'Dato entrante de puerto serial
5: dim BanderaLectura as byte      'Bandera de lectura realizada de dato entrante
6:
7: main:
8: ' Programa Principal
9: Ansel = 0                        'Inhabilita entradas analogas AN0-AN7
10: AnselH = 0                      'Inhabilita entradas analogas AN8-AN11
11: CM1CON0 = 0                    'Inhabilita comparador analogo 1
12: CM2CON0 = 0                    'Inhabilita comparador analogo 2
13: ADCON0 = 0                     'Inhabilita conversion analoga/digital 0
14: ADCON1 = 0                     'Inhabilita conversion analoga/digital 1
15: TrisA = 0                      'Configura Puerto A del PIC como salidas
16: TrisC = %11110000             'Configura bit 0-3 salidas y 4-7 entradas del puerto C
17: PortC = 0                     'Valor inicial de puerto C en 0
18: BanderaLectura = 0            'Valor inicial de variable en 0
19: Delay_ms(100)                  'Retardo para estabilizacion de configuraciones iniciales
20: UART1_Init(9600)               'Inicializa el modulo y establece la comunicacion a 9600 bps
21: Delay_ms(100)                  'Retardo para estabilizacion de modulo
22:
23: while (TRUE)                    'Inicio de ciclo secuencial
24:   if (UART1_Data_Ready() <> 0) then 'Si hay dato listo en puerto serial
25:     DatoEntrada = Uart1_Read()    'Guarda dato entrante en variable
26:     BanderaLectura = 1           'Coloca en 1 bandera de lectura
27:   end if                          'Fin de sentencia if
28:
29:   if BanderaLectura = 1 then      'Si bandera de lectura es 1
30:     select case DatoEntrada       'Seleccion de dato entrante
31:       case "0"                   'En caso es "0"
32:         portc.0 = 1              'Estado logico 1 bit 0 de puerto C
33:       case "1"                   'En caso es "1"
34:         portc.0 = 0              'Estado logico 0 bit 0 de puerto C
35:       case "2"                   'En caso es "2"
36:         portc.1 = 1              'Estado logico 1 bit 1 de puerto C
37:       case "3"                   'En caso es "3"
38:         portc.1 = 0              'Estado logico 0 bit 1 de puerto C
39:       case "4"                   'En caso es "4"
40:         portc.2 = 1              'Estado logico 1 bit 2 de puerto C
41:       case "5"                   'En caso es "5"
42:         portc.2 = 0              'Estado logico 0 bit 2 de puerto C
43:       case "6"                   'En caso es "6"
44:         portc.3 = 1              'Estado logico 1 bit 3 de puerto C
45:       case "7"                   'En caso es "7"
46:         portc.3 = 0              'Estado logico 0 bit 3 de puerto C
47:     end select                   'Fin de seleccion
48:     BanderaLectura = 0           'Coloca en 0 bandera de lectura
49:   end if                          'Fin de sentencia if
50:
51:   DatoEnviar = (Portc and %11110000) >> 4 'Entradas digitales a enviar
52:   UART1_WRITE(DatoEnviar)         'Se envia dato guardado en variable
53:   Delay_ms(100)                  'Retardo para que receptor lea dato
54: wend                               'Fin de ciclo secuencial
55: end.                               'Fin de programa
```

Fuente: elaboración propia.

5.3. Carga de programa en microcontrolador PIC16F690

Al momento de depurar y compilar el código en el IDE, este construye todos los archivos y configuraciones necesarios para que el lenguaje de bajo nivel que maneja el microcontrolador a través de sus registros internos sea interpretado correctamente y realice lo plasmado en el código.

5.3.1. Archivo hexadecimal

Este archivo es generado al momento de compilar el código en el IDE. Dicho archivo, por su extensión .HEX, indica que es el archivo en código hexadecimal, el cual es el producto final para ser cargado en la memoria del programa del microcontrolador.

Figura 17. Código de archivo hexadecimal

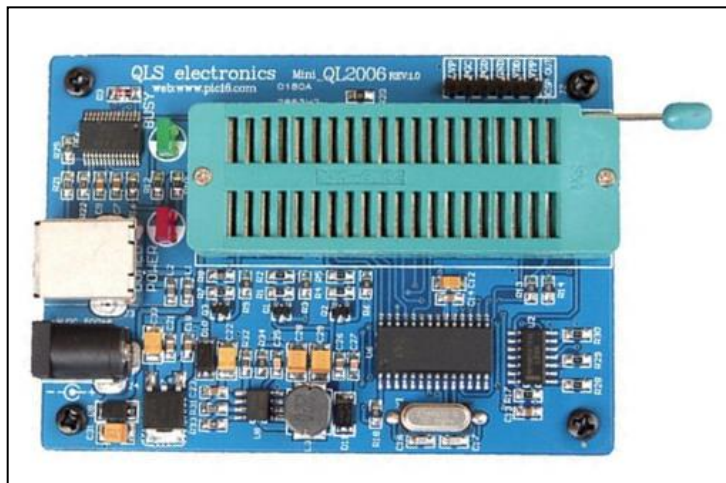
```
:020000003328A3
:100066000030831203138C1A0130F1007108F000DE
:020016000800E0
:100018008316031398169030831298008316861659
:10002800861383128C1E1B281A08F0001528080056
:10003800831203131A08F200981C242818121816A1
:0A0048007208F1007108F0000800D2
:100052008316031398182F280000292883122308D7
:0400620099000800F9
:10006600831203179E019F0199019A0103139F01B1
:1000760083169F018501F030870083128701A00156
:100086000330FB008A30FC005530FD00FD0B49288B
:10009600FC0B4928FB0B492800000008130831621
:1000A600990018150C200330FB008A30FC005530EF
:1000B600FD00FD0B5C28FC0B5C28FB0B5C2800009C
:1000C600000003207008003A03196E281C207008EF
:1000D600A2000130A0002008013A031DA32822082F
:1000E600303A031D78280714A2282208313A031D46
:1000F6007E280710A2282208323A031D8428871476
:10010600A2282208333A031D8A288710A22822082B
:10011600343A031D90280715A2282208353A031DF4
:1001260096280711A2282208363A031D9C2887150F
:10013600A2282208373A031DA2288711A228A00167
:10014600F0300705F2007208F000F00CF013F00C26
:10015600F013F00CF013F00CF0137008A100700807
:10016600A30029200330FB008A30FC005530FD0037
:10017600FD0BBB28FC0BBB28FB0BBB2800000000BB
:040186006428C428FD
:02400E00D20FCF
:00000001FF
```

Fuente: elaboración propia.

5.3.2. Carga de código hexadecimal en la memoria del programa del microcontrolador

Para esto es necesario contar con un programador del microcontrolador PIC de cualquier índole o marca que sea compatible. Para el caso del módulo *bluetooth* se ha utilizado el programador mini QL2006 de marca Qianlongsheng, junto con el software respectivo, el QL-Prog versión 2-36.

Figura 18. Programador QL2006



Fuente: PIC16. <http://pic16.com/>. Consulta: 23 de agosto de 2014.

Una vez realizada la carga del programa en memoria del microcontrolador, se debe colocar el PIC16F690 en la ranura respectiva en el módulo y suministrarle energía, con ello estará listo para ser utilizado en la aplicación que convenga. Cabe recalcar que aunque es un dispositivo para uso industrial, no limita usarse en aplicaciones o requerimientos no industriales, puesto que es flexible.

CONCLUSIONES

1. El dispositivo es bidireccional, puede ser utilizado en todos los casos como esclavo en la comunicación remota establecida, a través de comunicación serial, a una tasa de intercambio de 9 600 bits por segundo.
2. El diseño es orientado hacia el uso industrial, pero no lo limita al uso exclusivo en esta área, puesto que también puede ser utilizado en cualquier entorno no industrial.
3. La comunicación establecida entre el módulo y el equipo maestro es bidireccional, puede utilizarse sin problema en aplicaciones de automatización y sistemas de control en las que, por alguna u otra razón, es necesario utilizar comunicación inalámbrica fiable hasta distancias no mayores a 30 metros.
4. Todo equipo esclavo que desee conectarse al modulo deberá de tener servicio de comunicación serial a través de un puerto vinculado al transmisor/receptor *bluetooth* del maestro.
5. El modulo diseño es compatible con plataformas Windows, Linux, Android e iOS, lo que ofrece la posibilidad de entablar comunicación con una amplia variedad de dispositivos como computadoras, tabletas, teléfonos celulares, entre otros.

6. La opción para elegir el tipo de información a enviar o recibir depende únicamente de la aplicación realizada en el maestro y la tabla de actividades en función a la información recibida en el módulo.
7. El módulo en su construcción final tiene un costo de aproximadamente de 377 quetzales, equivalentes a unos 48 dólares, en comparación a referencias de equipos similares con costos de 84 dólares, 43 % más económico que sus similares más baratos.
8. Es un diseño sencillo que permite el mantenimiento o reparación en corto tiempo y a bajo costo.
9. Al estar operando más de uno de estos módulos dentro de la misma área, no se interfieren entre sí, ya que cuentan con autenticación independiente entre enlaces de comunicación y la banda en la cual trabajan es multiplexada en el tiempo y frecuencia.

RECOMENDACIONES

1. Si se desea utilizar como base el dispositivo planteado en el presente trabajo, para realizar un diseño propio, se debe implementar una pantalla de visualización que provea información como establecimiento de comunicación, actividad de las salidas y estado de las entradas digitales. De igual manera, contemplar el uso de un microcontrolador más grande para obtener mayor cantidad de entradas y salidas digitales.
2. Para distancias mayores a 30 metros, lo cual está limitado por la versión del *bluetooth* implementado en el módulo, es necesario utilizar una versión superior, como la versión 4,0 de *bluetooth* para alcanzar distancias mayores, siempre y cuando el maestro tenga la misma versión de *bluetooth* que el módulo.
3. Para transmisiones a tasas mayores o menores a 9 600 baudios, el módulo debe ser configurado con la nueva tasa de transmisión deseada a través de comandos AT detallados por el fabricante del dispositivo.

BIBLIOGRAFÍA

1. ARCHUNDIA PAPACETZI, Francisco Martín. *Wireless personal area network (WPA) & home networking*. [en línea]. <http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/>. [Consulta: 30 de mayo de 2015].
2. BAQUERO, Chistian. *Análisis comparativo del marketing de proximidad vs. técnicas tradicionales de marketing*. [en línea]. <<http://repositorio.puce.edu.ec/handle/22000/3760>>. [Consulta: 14 de abril de 2013].
3. BLUETOOTH BLOG. *La casa inteligente bluetooth*. [en línea]. <<http://blog.bluetooth.com/>>. [Consulta: 15 de marzo de 2015].
4. BLUETOOTH. *¿Qué es la tecnología bluetooth?* [en línea]. <<http://www.bluetooth.com/>>. [Consulta: 15 de enero de 2015].
5. CASTELA, Pedro J. *El relé*. [en línea]. <<http://platea.pntic.mec.es/~pcastela/tecno/documentos/apuntes/rle.pdf>>. [Consulta: 20 de marzo de 2015].
6. Comunicación entre Dispositivos Bluetooth. *Arquitectura de Bluetooth*. [en línea]. <<http://ulfsark.sourceforge.net/ulfsarkdocweb.pdf> >. [Consulta: 12 de diciembre de 2014].

7. Estándares IEEE. *Estandar IEEE 802.15.1.2002* [en línea].
<<https://standards.ieee.org/findstds/standard/802.15.1-2002.html>>.
[Consulta: 12 de junio de 2015].
8. FERNÁNDEZ GÓMEZ, Eva. *Conocimientos y aplicaciones tecnológicas para la dirección comercial.* [en línea].
<<https://books.google.com.gt/books?id=JvXbRLUpZiwC&pg=PA39&dq=TECNOLOGIA+BLUETOOTH&hl=es&sa=X&ved=0CCkQ6AEwAmoVChMIsMLOz8qxxwIViyseCh2zlg0c#v=onepage&q&f=false>>
[Consulta: 30 de abril de 2015].
9. GIRONA SALGADO, Jordi. *Principio de comunicaciones móviles.* [en línea].<<https://books.google.com.gt/books?id=GAf5BKJOfUwC&pg=PA41&dq=TECNOLOGIA+BLUETOOTH&hl=es&sa=X&ved=0CCIQ6AEwAWoVChMIsMLOz8qxxwIViyseCh2zlg0c#v=onepage&q&f>>.
[Consulta: 20 de marzo de 2015].
10. HUIDOBRO MOYA, José Manuel. *Sistemas telemáticos.* [en línea].
<<https://books.google.com.gt/books?id=epcG9USnmqUC&pg=PA116&dq=TECNOLOGIA+BLUETOOTH&hl=es&sa=X&ved=0CCcQ6AEwAzgKahUKEwjzNfz0LHHAhVFHB4KHe>>. [Consulta: 30 de abril de 2015].
11. IEEE. Tutoriales IEEE 802.15. *Guía web.* [en línea].
<<http://www.ieee802.org/15/pub/Tutorials.html>>. [Consulta: 15 de febrero de 2015].

12. Introducción a MikroBasic. *Características principales del lenguaje de programación basic.* [en línea]. <<https://sites.google.com/site/cursopicmikrobasic/introduccion-a-mikrobasic>>. [Consulta: 10 de enero de 2015].
13. Microchip Technology Inc. *Hoja de datos PIC16F690.* [en línea]. <ww1.microchip.com/downloads/en/DeviceDoc/41262A.pdf>. [Consulta: 16 de mayo de 2015].
14. National Instruments. *Comunicación serial: conceptos generales.* [en línea]. <<http://digital.ni.com/public.nsf>>. [Consulta: 12 de septiembre 2014].
15. STATISTA. *Sitio oficial de Statista* [en línea]. <<http://www.statista.com/statistics/283638/installed-base-forecast-bluetooth-enabled-devices-2012-2018/>>. [Consulta: 22 de mayo de 2015].

