



Universidad de Mendoza

Facultad de Ingeniería

Ingeniería en Computación

“Sistema de Domótica comandado por voz para
personas con discapacidad motriz y visual utilizando
Redes ZigBee”

Autor: Guillermo Paladini

Asesor: Ing. Lucas Iacono

Año: 2014

Mendoza - Argentina

AGRADECIMIENTOS

Haber llegado a la meta luego de haber transitado un camino colmado de dificultades no hubiese sido posible sin la ayuda de la gente que me rodea.

Agradecer a Dios, por la familia que me ha dado, incondicional y afectuosa en todo momento de dificultad.

A mis padres, fuentes de toda inspiración, amor y apoyo, con su ejemplo me han llevado a aprender que en la vida las cosas se logran con sacrificio y esfuerzo.

Al Ing. Lucas Iacono, por su orientación técnica y su ayuda permanente durante mi carrera y en especial, para finalizar este trabajo.

A mis amigos, por tener siempre las palabras de apoyo necesarias, por brindar siempre su ayuda incondicional.

Gracias.

ÍNDICE

RESUMEN	6
CAPÍTULO I	
1. INTRODUCCIÓN	8
1.1 OBJETIVOS	10
CAPÍTULO II	
2. MARCO TEÓRICO	11
2.1. ESTADO ACTUAL DEL TEMA	11
2.2. DESCRIPCIÓN DE LOS ELEMENTOS TEÓRICOS	12
2.2.1. DOMÓTICA	12
2.2.2. MICROCONTROLADORES	13
2.2.3. SENSORES Y ACTUADORES	14
2.2.3.1. SENSOR O TRANSDUCTOR DE ENTRADA	14
2.2.3.1.1. TIPOS DE SENSORES	15
2.2.3.2. ACTUADORES	16
2.2.3.3. SMART SENSOR – NODOS SENSORES	16
2.2.4. COMUNICACIÓN INALÁMBRICA	17
2.2.5. TECNOLOGÍA INALÁMBRICA ZIGBEE	18
2.2.5.1. CONSUMO DE POTENCIA Y AHORRO DE ENERGÍA	19
2.2.5.2. DISPOSITIVOS ZIGBEE	21
2.2.5.2.1. COORDINADOR	21
2.2.5.2.2. ENRUTADOR	21
2.2.5.2.3. DISPOSITIVOS FINALES	22
2.2.5.2.4. FFD (DISPOSITIVO DE FUNCIÓN COMPLETA)	22
2.2.5.2.5. RFD (DISPOSITIVO DE FUNCIÓN REDUCIDA)	22
2.2.5.3. ARQUITECTURA DE RED ZIGBEE	23
2.2.5.3.1. TOPOLOGÍAS	24
2.2.5.3.2. CAPA FÍSICA 802.15.4	27
2.2.5.3.2.1. BANDAS DE FRECUENCIA Y CANALES DE OPERACIÓN	28
2.2.5.3.2.2. DISTANCIA DE COBERTURA	29
2.2.5.3.3. CAPA DE ENLACE 802.15.4	29

2.2.5.3.3.1. ASOCIACIÓN Y DISOCIACIÓN	31
2.2.5.3.3.2. ASIGNACIÓN DE DIRECCIONES	32
2.2.5.3.4. CAPA DE RED ZIGBEE	32
2.2.5.3.4.1. NLDE – ENTIDAD DE DATOS DE LA CAPA DE RED	33
2.2.5.3.4.2. NLME – ENTIDAD DE GESTIÓN DE LA CAPA DE RED	33
2.2.6. RECONOCIMIENTO DE VOZ	35
2.2.6.1. TÉCNICAS DE RECONOCIMIENTO	40
2.2.7. SERVICIO DE GOOGLE PARA EL RECONOCIMIENTO DE VOZ.	43
2.2.7.1. MÉTODOS DE RAH IMPLEMENTADOS POR GOOGLE	43
2.3. METODOLOGÍAS DE INVESTIGACIÓN	44
2.3.1. HIPÓTESIS	44
2.3.2. TÉCNICAS DE RECOLECCIÓN DE DATOS	45

CAPÍTULO III

3. DESCRIPCIÓN DEL SISTEMA	46
3.1. DESCRIPCIÓN GENERAL DEL SISTEMA	46
3.2. DESCRIPCIÓN DEL HARDWARE	49
3.2.1. COORDINADOR	49
3.2.2. DISPOSITIVOS FINALES O NODOS	51
3.3. DESCRIPCIÓN DEL SOFTWARE DE RECONOCIMIENTO DE VOZ	53
3.4. DESCRIPCIÓN DEL FIRMWARE	55
3.4.1. FIRMWARE DEL COORDINADOR	55
3.4.2. FIRMWARE DE LOS DISPOSITIVOS FINALES O NODOS	57

CAPÍTULO IV

4. DESARROLLO DE INGENIERÍA	59
4.1. DESARROLLO DEL HARDWARE	59
4.1.1. DESARROLLO DEL DISPOSITIVO COORDINADOR	59
4.1.2. DESARROLLO DE LOS DISPOSITIVOS FINALES	61
4.2. DESARROLLO DEL FIRMWARE	63
4.2.1. FIRMWARE DEL COORDINADOR	63
4.2.2. FIRMWARE DE LOS DISPOSITIVOS FINALES	65
4.3. DESCRIPCIÓN FUNCIONAL DEL SISTEMA	67

4.3.1. FUNCIONAMIENTO	67
CAPÍTULO V	
5. ANÁLISIS DE COSTOS	70
CAPÍTULO VI	
6. CONCLUSIONES	73
CAPÍTULO VII	
BIBLIOGRAFÍA	75
ANEXOS	79

RESUMEN

El objetivo del presente trabajo final es el diseño y construcción de un prototipo domótico adaptado a las necesidades de una persona con discapacidad motriz y visual.

En primer lugar, se investiga la tecnología del estándar ZigBee para redes de área personal inalámbricas (WPAN). Dicho estándar ha sido desarrollado para usarse en la automatización de hogares, entre otras aplicaciones. Además se estudia la tecnología de Reconocimiento Automático del Habla para controlar el sistema domótico a implementar.

Una vez diseñada la red de sensores y actuadores, se adquieren los módulos ZigBee y los demás componentes electrónicos para comenzar a desarrollar y construir los nodos de sensores y actuadores, y el coordinador de la PAN. Finalizado este proceso se da paso al ensamblaje final del sistema.

Las conclusiones demuestran que el sistema genera un aumento en la calidad de vida de las personas a las cuales estaba destinado.

Se comprueba también que el estándar ZigBee es ideal para la creación de redes inalámbricas y el manejo de dispositivos asociados a sus nodos.

1. INTRODUCCIÓN

La utilización de la tecnología para mejorar la calidad de vida de las personas y facilitar sus tareas domésticas, comerciales, profesionales e industriales, ha demostrado grandes avances. Es evidente que la ciencia de la computación y la electrónica están cada vez más presentes en nuestra vida cotidiana.

Pero al día de hoy, aunque la tecnología aparenta estar al alcance de todos, se ven necesidades no cubiertas en algunos sectores sociales. Este es el caso de las personas con discapacidad motriz o visual. Discapacidades que por sus características, dificultan enormemente la movilidad y el diario desenvolvimiento del individuo que la padece, limitando su calidad de vida.

En este trabajo se adaptará un sistema de domótica, al hogar de una persona con discapacidad, para que ésta pueda desenvolverse en sus quehaceres diarios con mayor facilidad y seguridad, brindándole todas las ventajas que la domótica ofrece, y controlando el sistema mediante el uso de su voz.

La Real Academia Española, define en la vigésima segunda edición de su diccionario, a la Domótica como el ***“Conjunto de sistemas que automatizan las diferentes instalaciones de una vivienda”***. Según la misma fuente, el origen de la palabra Domótica, viene del Latín, “Domus” (Casa) e informática.

Además, la enciclopedia Larousse define la domótica como **concepto de vivienda que integra todas las funciones en materia de seguridad,**

gestión de energía, facilidades de comunicación, control de automatismos.

Se puede entonces proponer una definición más técnica y completa de Domótica, diciendo que es **Un conjunto de sistemas electrónicos, instalados en una vivienda con el objetivo de automatizar las tareas cotidianas, ofreciendo confort y comodidad, proveer más seguridad controlando las alarmas y simulando la presencia de una persona cuando el hogar se encuentre desocupado, ahorrando además energía mediante el control del sistema de calefacción e iluminación.**

Para esto se recurrirá al uso de distintas técnicas y tecnologías como la programación de microcontroladores para la automatización de sistemas de control y el estándar ZigBee para Wireless Personal Area Network, (WPAN), estándar que por sus características resulta ideal para implementar la comunicación entre los distintos dispositivos que serán controlados en un sistema domótico, y un sistema de reconocimiento de voz para que la persona pueda manejar el sistema.

El presente trabajo final estará estructurado de la siguiente manera. En el Capítulo I se presentan los objetivos. Luego en el Capítulo II se desarrolla el marco teórico, presentando el estado actual y la descripción de las tecnologías que se usarán. El Capítulo III describe el funcionamiento general del sistema así como también cada uno de los subsistemas que lo componen. El Capítulo IV de explica y describe el desarrollo de cada parte de las 3 partes que componen el sistema, Software, Firmware, Hardware.

1.1 Objetivos

Objetivos Generales

- Construir un sistema de domótica adaptado a la situación particular de una persona con discapacidad motriz o visual, con el fin de mejorar su calidad de vida y proporcionarle mayor seguridad en su diario desempeño.
- Profundizar en el estándar ZigBee, como herramienta de control de equipos a distancia.

Objetivos o metas específicas

- Obtener las necesidades de una persona con discapacidad motriz o visual para planificar que tareas deberán automatizarse.
- Desarrollar el hardware y el software de los nodos de red que controlaran los dispositivos a automatizar.
- Desarrollar el hardware y el software del sistema embebido que se utilizará para coordinar el funcionamiento del sistema domótico.
- Mejorar la seguridad y calidad de vida de la persona con discapacidad motriz o visual.

2. MARCO TEORICO

2.1. Estado actual

En la actualidad, la tecnología de la domótica se ha difundido en la sociedad, brindando seguridad, comodidad, mejor calidad de vida y ahorro de recursos energéticos

En el caso de personas con discapacidad lo más importante que les proporciona la domótica, son los sistemas de control de los elementos de su entorno. De esta forma, pueden desenvolverse por ellas mismas tanto en el hogar como en el puesto de trabajo, sin la necesidad de contar con terceras personas que las asistan permanentemente.

Hoy en día ya se cuentan con diferentes tipos de accesorios diseñados especialmente para la ayuda de personas con problemas de discapacidad. Dichos accesorios han sido diseñados para dar solución a un problema en específico y al mismo tiempo ser totalmente integrables para dar solución completa a una problemática, en casos de movilidad reducida severa se pueden encontrar mecanismos habilitados para levantarse de la cama, acceder a la ducha, mandos para abrir puertas, etc. [9]

Por otro lado, la domótica permite que una persona con discapacidad o movilidad reducida pueda encender y apagar sus aparatos electrónicos mediante la voz, mandos a distancia, pantallas táctiles o a través de Internet; incluso les permite realizar llamadas de emergencia o activar sus servicios de alarma si lo necesitan.

Otro avance importante de la tecnología es la tele-asistencia que permite a las personas de edad avanzada o con algún tipo de discapacidad vivir de forma más autónoma, debido a que los encargados de cuidarlos pueden monitorear su actividad desde puntos remotos, utilizando Internet, telefonía móvil, etc. [9]

2.2. Descripción de los elementos Teóricos

2.2.1. Domótica

En términos generales, se puede definir la Domótica como la tecnología encargada de desarrollar e implantar la automatización de las instalaciones habituales en una vivienda o edificio. [1]

Un sistema domótico tiene como objetivos el accionamiento y control de los artefactos electrónicos, iluminación, riego de jardines, cerraduras eléctricas. La domótica permite también, controlar la seguridad en el hogar.

El sistema debe permitir el monitoreo constante del hogar donde esté instalado. Ante la ocurrencia de un accidente el sistema debe emitir una alarma y notificar a las personas indicadas como responsables, o al centro de mando que brinde el servicio de monitoreo.

Dentro de esta rama de la tecnología encontramos dos ramas que son: las viviendas inteligentes y la domótica aplicada a la accesibilidad. Respecto a la domótica aplicada a la accesibilidad, consiste en aumentar la autonomía de las personas con discapacidad. [10]

2.2.2. Microcontroladores

Un Microcontrolador, es una herramienta de electrónica integrada que cuenta con los 3 elementos básicos de una computadora (Unidad central de Proceso, Memoria, Puertos de Entrada y Salida).

Poseen memorias EEPROM (Memoria Programable y electrónicamente borrable), como también memoria RAM (memoria de trabajo de acceso aleatorio). Hoy en día, los microcontroladores están equipados con unidades periféricas como puertos serie (UART, I2C, SPI, USB), Conversores analógicos-Digitales, Puertos Ethernet, etc.

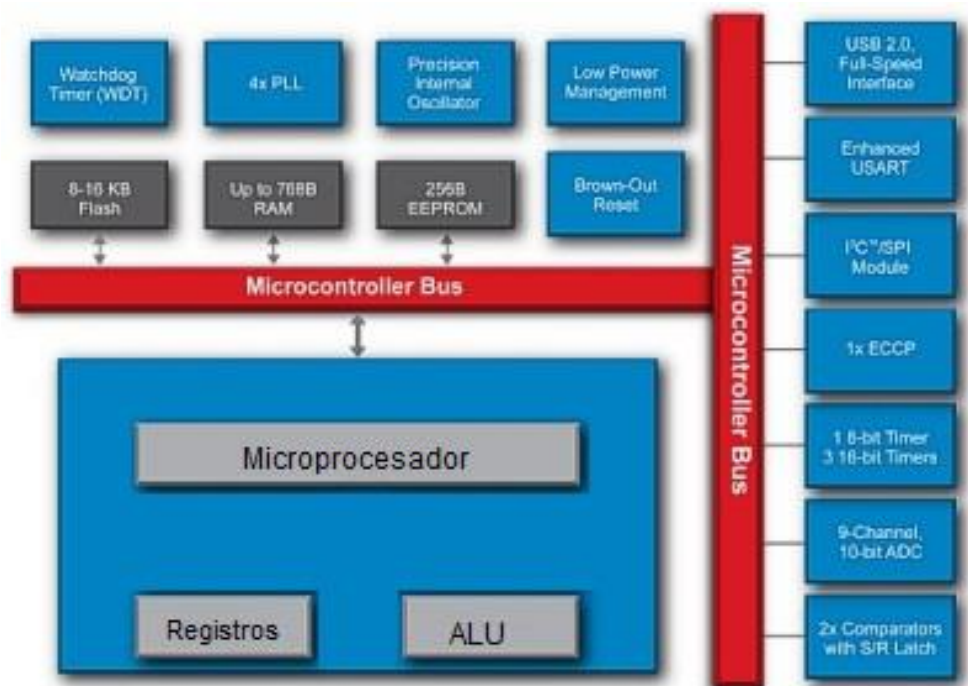


Figura 1- Partes de un Microcontrolador.

El uso de los microcontroladores está orientado a aplicaciones dedicadas a tareas específicas, a diferencia de los microprocesadores de propósito general, que son usados en las Computadoras Personales (o PC) y que requieren mayor potencia de procesamiento.

Un microcontrolador se usa para tareas de automatización y control de dispositivos electrónicos, electromecánicos u ópticos, algunos ejemplos de estas aplicaciones son los sistemas de control electrónico de motores de Corriente continua, Servomotores y motores paso a paso, dispositivos de bioingeniería y medicina, controles remotos, máquinas de oficinas, herramientas eléctricas, y juguetes.

La elección de un microcontrolador debe hacerse en forma cuidadosa, basándose en la cantidad de potencia de proceso necesitada, cantidad de puertos de E/S requeridos, costo, tamaño, etc.

2.2.3. Sensores y Actuadores

2.2.3.1. Sensor o transductor de entrada

Un **sensor** es un dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas. Las variables de instrumentación pueden ser por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, pH, etc. [2]

Dichas señales, deben ser acondicionadas previamente para poder ser interpretadas por el elemento de control.

Su construcción está basada en los fenómenos Físicos y Químicos, y en los materiales en los cuales dichos fenómenos se manifiestan de una forma eficaz y eficiente para su medición, es decir, con sensibilidad, repetitividad y especificidad suficientes. Para el funcionamiento de los

sensores, interesan los efectos causados en las propiedades eléctricas de los materiales con los que fue construido, por ejemplo su conductividad eléctrica.

Se pueden clasificar los materiales con los que se construyen los sensores en: Conductores, semiconductores y dieléctricos. También se encuentran sensores fabricados con materiales magnéticos, cuyas propiedades son utilizadas para medir cambios en la magnitud, dirección, y otras propiedades de los campos magnéticos.

Las variables medidas tienen el nombre de “variables de instrumentación” y ellas pueden ser, por ejemplo: temperatura, intensidad lumínica, humedad, sonido, etc. A continuación se especifican los tipos de sensores que se encuentran en el mercado.

2.2.3.1.1. Tipos de Sensores

Existe una muy amplia gama de dispositivos transductores de entrada (o sensores) que son de uso comercial, cada uno de ellos cumple una función de medición o detección específica. Estos pueden ser:

- Mecánicos: Servoacelerómetros, detectores de posición.
- Hidráulicos: Galgas extensiométricas, diafragmas.
- Ópticos: Detectores de Luz UV o IR, foto-diodos.
- Térmicos: Termopar, Termocupla, Humedad.
- Magnéticos: Inductivos, Efecto Hall.
- Eléctricos: Sensores electrostáticos
- Moleculares o Químicos

2.2.3.2. Actuadores o Transductores de Salida

Son elementos que realizan una acción y provocan un efecto sobre un proceso automatizado. Estos dispositivos, capaces de generar una fuerza mecánica o eléctrica, son los utilizados para llevar a cabo las órdenes que provienen del controlador. El actuador da una salida necesaria para activar a un elemento final de control, que puede ser un brazo robótico, una serie de engranajes, una bomba de extracción de líquidos, etc.

Existen varios tipos de actuadores como ser:

- Electrónicos y eléctricos
- Hidráulicos– Neumáticos

2.2.3.3. SMART Sensors– Nodos Sensores

Los grandes avances en las tecnologías de los semiconductores y circuitos integrados, ha permitido desarrollar microcontroladores de bajo costo y grandes escalas de integración. Hoy en día es posible juntar en un mismo integrado el dispositivo sensor o transductor y la electrónica necesaria para tomar decisiones. A este elemento se lo denomina “smart-sensor”.

Físicamente un smart-sensor consiste de un elemento transductor de entrada, electrónica de acondicionamiento de señal, y un controlador que pueda tomar ciertas decisiones en base a la información que recibió. [6]

Se puede entonces definir smart-sensor como: sensores con electrónica integrada que puede efectuar las siguientes funciones [6]:

- Funciones Lógicas
- Comunicación de dos vías
- Toma de Decisiones

Si se dota a este tipo de sensores con un módulo de comunicaciones inalámbrico con capacidad de conectarse a una red, tenemos en conjunto lo que Raghunathan et. Al. [7] denominan nodo sensor.

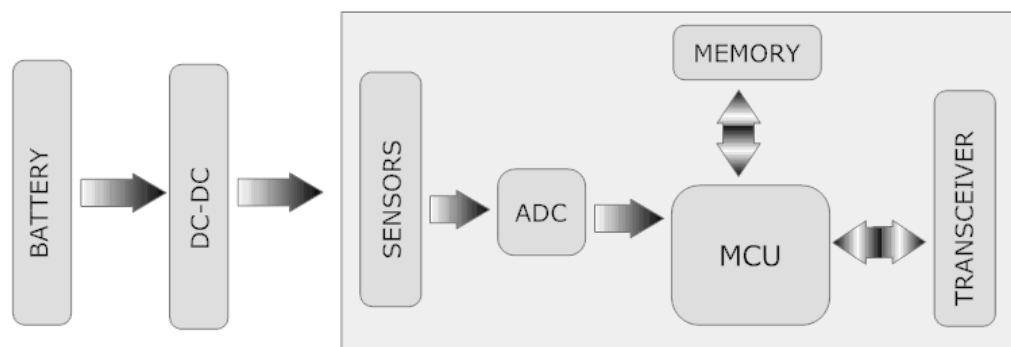


*Figura 2 – Diagrama en Bloques de un Smart Sensor
Raghunathan, Schurgers, Park, Srivastava*

2.2.4. Comunicación Inalámbrica

Una red inalámbrica permite la transferencia de datos y las comunicaciones por medio de ondas electromagnéticas, que se propagan en el vacío, evitando así tener que realizar la instalación de cables de cobre, ni afectar la estética de una vivienda.

En la actualidad existen distintos estándares que definen la utilización de redes inalámbricas; Wi-Fi, Bluetooth, ZigBee entre otros.

WiFi

El Wi-Fi, cuya base es el estándar IEEE 802.11, es utilizado ampliamente en todo el mundo para la creación de redes de área local inalámbricas, en oficinas, comercios y hogares, también en industrias y

para la conexión de puestos remotos a donde se haría muy difícil llegar a través de cables de cobre o fibra óptica. Wi-Fi opera en la banda de 2.4 GHz, aunque hay una especificación, la 802.11n, que lo hace también en la banda 5.8 GHz.

Si la red inalámbrica permite la transferencia de datos en un área pequeña acotada a un hogar, esta red se denomina Red Inalámbrica de Área Personal o (WPAN Wireless Personal Area Network).

Bluetooth

El Bluetooth fue creado como un estándar para redes inalámbricas de área personal, es el estándar 802.15.1 de la IEEE. Comenzó a desarrollarse en 1998, luego de que Ericsson, Nokia, IBM, Toshiba e Intel crearan un grupo de interés para promover su desarrollo. Bluetooth, está ubicado en la banda de frecuencias de 2.4 GHz, provee entre otras características seguridad y robustez. Es ampliamente utilizado en Teléfonos celulares, PDA's, Impresoras y vehículos. Una limitación importante de la tecnología Bluetooth es la acotada cantidad de dispositivos o nodos que pueden pertenecer a la misma red (en Bluetooth llamada PICONET), ya que el máximo es solo de 8.

2.2.5. Tecnología Inalámbrica ZigBee

ZigBee es un estándar, desarrollado por la *ZigBee Alliance* [3], que proporciona una solución para comunicaciones inalámbricas de bajo costo y consumo.

El estándar ZigBee, está basado en el nivel físico y en el control de acceso al medio, definidos en la versión 2003 del estándar IEEE 802.15.4, que desarrolla estos niveles para redes de área personal (PAN) de baja tasa de transferencia y bajo consumo. Al igual que Wi-Fi y Bluetooth, el estándar ZigBee trabaja en el rango de frecuencias de 2.4GHz, permitiendo la utilización de 16 canales, cada uno de 5 MHz de ancho de banda.

Es importante mencionar que en el año 2007, la ZigBee Alliance desarrollo el perfil “ZigBee Home Automation - ZHA”, diseñado con el objetivo de definir prácticas estándar para la automatización de hogares y pequeños comercios u oficinas; y con el propósito de permitir la operatividad entre dispositivos ZHA producidos por distintos fabricantes. Motivo por el cual, ZigBee resulta una buena alternativa para la construcción de un sistema Domótico.

2.2.5.1. Consumo de Potencia y Ahorro de Energía

Una de las principales ventajas del estándar ZigBee es que ha sido diseñado para bajas tasas de transferencia, y por lo tanto, para bajo consumo de potencia y ahorro energético. Los dispositivos ZigBee están diseñados para funcionar alimentados solo con baterías y tener una autonomía de varios años sin necesidad de reemplazar la fuente de alimentación. Esto se logra dado que estos dispositivos permiten cambios de estado de “activo” a “sleep”.

En estado activo, un dispositivo ZigBee usualmente consume 24.7 mA en la Transmisión (Tx) y 27 mA en la Recepción (Rx), para un voltaje de alimentación de 3 V.

Un gráfico del estudio de Jin-Shyan Lee, Yu-Wei Su y Chung-Chou Shen [5], nos muestra una comparación de la potencia consumida entre los cuatro estándares mencionados en su estudio.

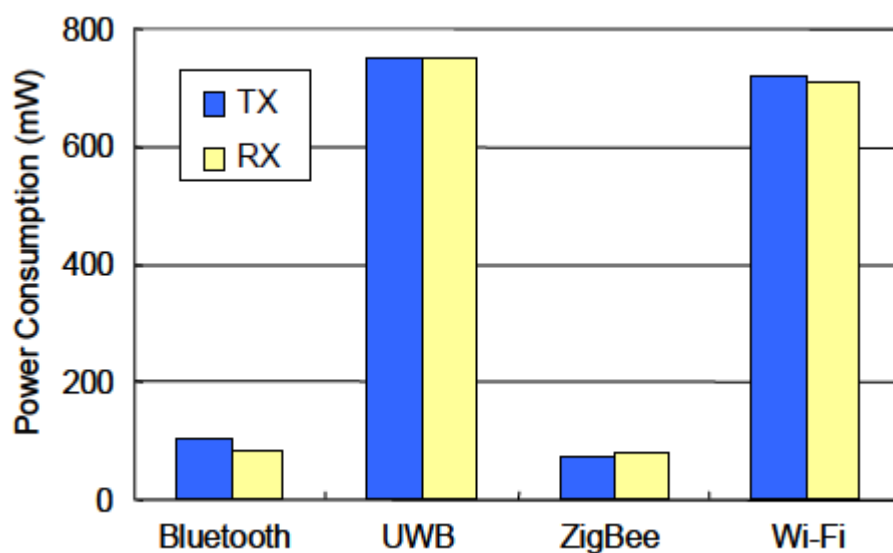


Figura 3-Comparación de la potencia consumida entre los cuatro estándares

Se puede observar que tanto en Tx como en Rx, el consumo de potencia de los dispositivos ZigBee es muy bajo, comparado con otros estándares como Wi-Fi y UWB.

2.2.5.2. Dispositivos ZigBee

ZigBee define tres tipos de dispositivos: Coordinador, Router y End Devices.

El coordinador y el router son Fully Function Device (FFD) según el estándar 802.15.4 y El End Device es un Reduced Function Device según 802.15.4.

FFD y RFD son explicados en los puntos 2.2.5.2.4 y 2.2.5.2.5 respectivamente

2.2.5.2.1. Coordinador

El tipo de dispositivo más completo. Debe existir uno por red. Sus funciones son las de iniciar la red (selecciona el PAN ID y el canal de transmisión). Asociar a los distintos dispositivos y administrar el tráfico de datos dentro de la red

2.2.5.2.2. Enrutador

Es un dispositivo ZigBee cuya función es armar redes tipo mesh. Mantiene una tabla de ruteo y controla la asociación y disociación de los dispositivos ZigBee End Devices que tiene asignados. Cuando un requerimiento es recibido desde un router que no está presente en la tabla de ruteo, el router enviara una solicitud de ruteo, la cual es transmitida por toda la red. Cuando el dispositivo objetivo recibe la solicitud, este enviara el camino que tenga menor costo o el camino más corto para llegar al nodo objetivo.

2.2.5.2.3. Dispositivos Finales

Posee la funcionalidad necesaria para comunicarse con su nodo padre (el coordinador o un router), pero no puede transmitir información destinada a otros dispositivos. De esta forma, este tipo de nodo puede estar dormido la mayor parte del tiempo, aumentando la vida media de sus baterías. Un ZED tiene requerimientos mínimos de memoria y es por tanto significativamente más barato.

2.2.5.2.4. FFD – Dispositivo de Funcionalidad Completa

Es un dispositivo ZigBee de gran capacidad de procesamiento. Puede adoptar cualquiera de las tres topologías que define el stack y debido a sus características puede ser definido como Coordinador de la red, o como router. Puede comunicarse con cualquier otro dispositivo (RFD o FFD) e implementa el set de protocolo completo. Debido a sus características es un dispositivo más costoso que el RFD.

2.2.5.2.5. RFD – Dispositivo de Funcionalidad Reducida

Es un dispositivo de menor capacidad de procesamiento, y por lo tanto menor costo. Está limitado a ser un dispositivo final en una red de topología estrella o de malla. Debido a su limitada capacidad de procesamiento, no puede ser coordinador de red ni enrutador en una red ZigBee. Es de muy sencilla implementación y contiene un set de protocolos reducido.

2.2.5.3. Arquitectura de red ZigBee

La arquitectura del estándar ZigBee está construida a partir de un conjunto de bloques llamados “layers” o capas. Cada capa cumple una función determinada en el estándar y provee un servicio a la capa de más alto nivel a través de los SAP, “Service Acces Point” o Puntos de Acceso de Servicio.

Una Data Entity (Entidad de Datos) provee el servicio de transmisión de datos, y la Management Entity (Entidad de Gestión) provee los demás servicios requeridos. Cada una de estas entidades (de datos y de manejo) dispone de una interface para comunicarse con la capa superior a través de los ya mencionados SAP, y cada SAP cuenta con un número de primitivas de servicio para lograr la funcionalidad requerida.

El estándar 802.15.4 define dos capas, la física (PHY) y la sub-capa de control de acceso al medio (MAC). ZigBee se basa en 802.15.4 y aporta por sobre estas la capa de red (NWK) y el marco necesario para la capa de aplicación (APL). [3]

Dicho marco consiste de una APS “Aplication Support Sub-layer” o Sub-capa para soporte de aplicación, y del ZDO o ZigBee Device Object.

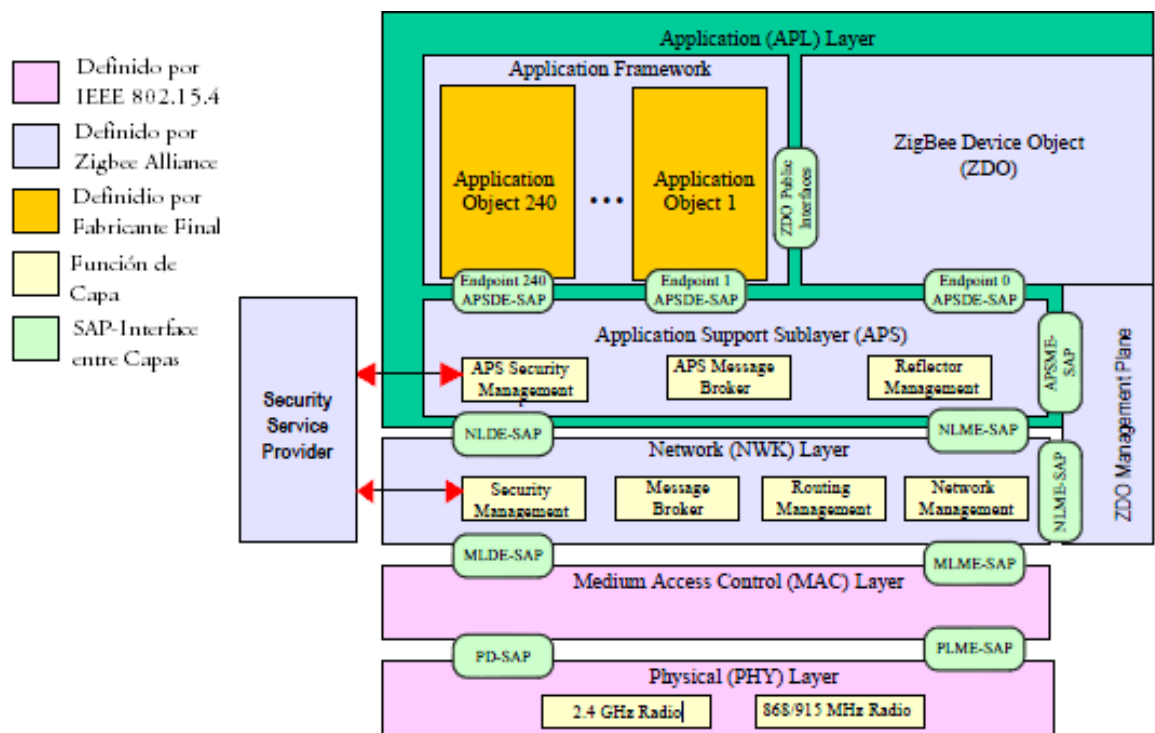


Figura 4 – Diagrama en Bloques del Stack ZigBee

2.2.5.3.1. Topologías

Una red de dispositivos ZigBee puede utilizar cuatro tipos de topologías, dos de ellas están definidas en el estándar IEEE 802.15.4 y son la topología de estrella y topología punto a punto. Las dos restantes, topología de malla (tipo mesh) y topología árbol, las define el estándar ZigBee en su capa de Red, gracias a que esta agrega la posibilidad de utilizar dispositivos ZigBee como routers.

En una **topología de Estrella** la red es controlada por un dispositivo único llamado “Coordinador”, y tiene asociados otros dispositivos finales. La función del coordinador es iniciar y finalizar la comunicación, agregar o quitar dispositivos a la red. Los dispositivos finales, tienen alguna función particular y son el punto de inicio o de fin de una comunicación. El

coordinador también puede ser usado para comunicar mensajes desde un dispositivo final a otro de su mismo tipo.

Para la **topología punto a punto**, al igual que la topología estrella, es controlada por un Coordinador, pero a diferencia de esta, cualquier dispositivo puede comunicarse con otro siempre y cuando estén en el mismo rango de alcance circundante.

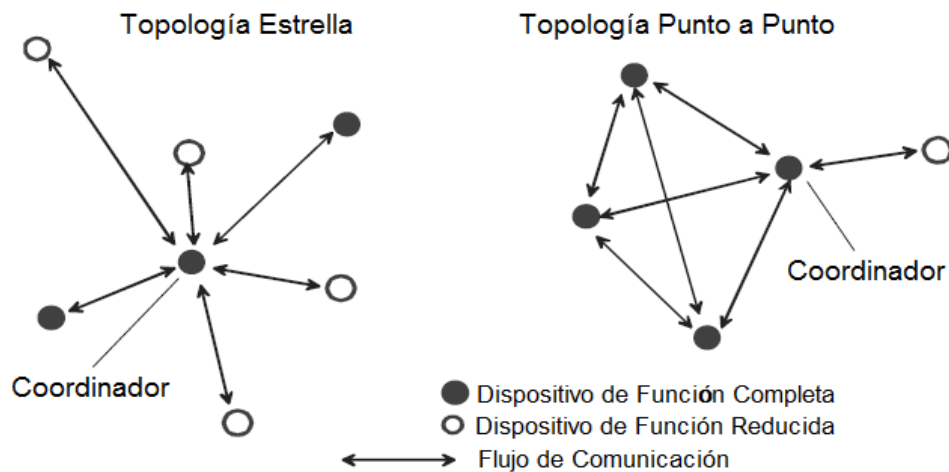


Figura 5 – Diagramas de topologías Estrella y Punto

En la **topología de Malla o Mesh** un nodo opera como coordinador de red sincronizando al resto de los nodos. La topología de malla se forma a través del uso de routers ZigBee. El algoritmo de encaminamiento utiliza un protocolo de pregunta-respuesta (request-response) para eliminar las rutas que no sean óptimas.

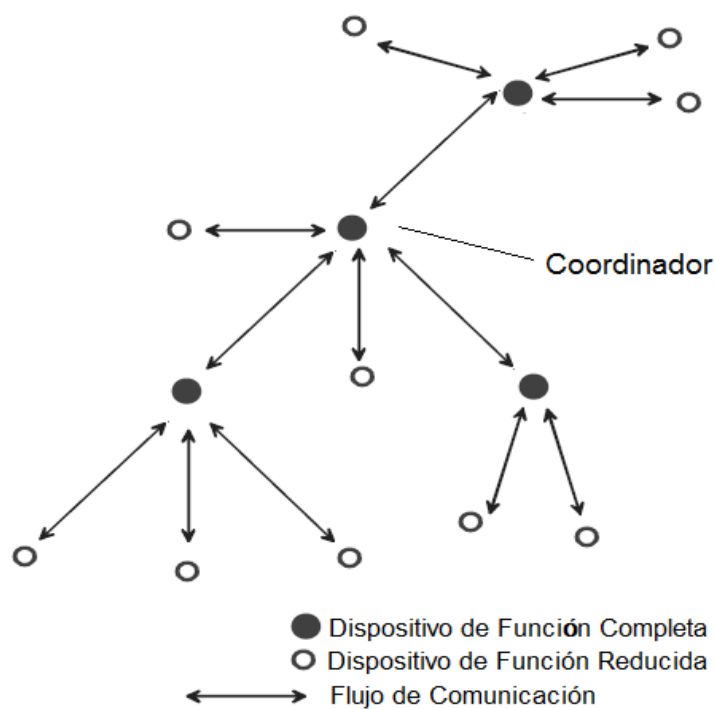


Figura 6 – Diagrama de Topología Mesh

La **topología de Árbol** emplea routers para mover los mensajes de datos y de control a través de la red usando una estrategia de enrutamiento jerárquica. Los routers cumplen la función de re-transmitir los mensajes a los dispositivos finales, que están fuera del alcance del coordinador de la Red.

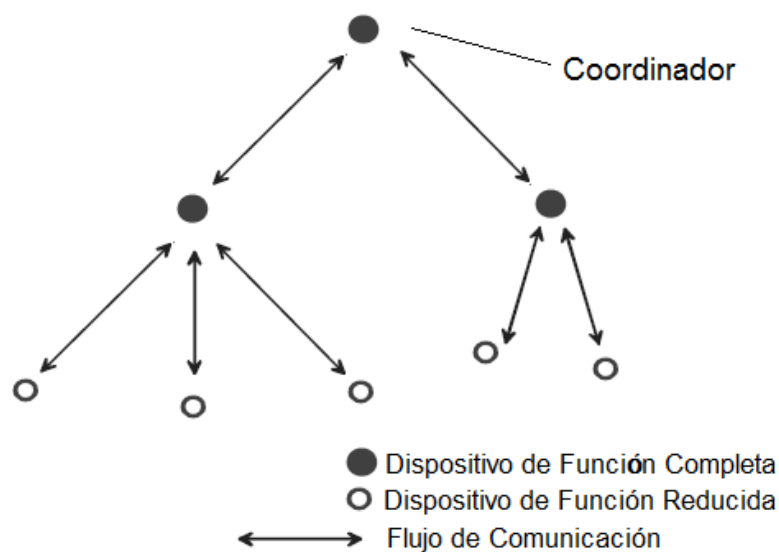


Figura 7 – Diagrama de Topología Árbol

2.2.5.3.2. Capa Física

La función de esta capa es proveer de una interface entre el canal de radio físico y la sub-capa MAC. Las principales características funcionales de esta capa son:

- Activar y Desactivar el Transmisor-receptor de radio
- Seleccionar el Canal
- Detección de Energía en el Medio
- Indicar Calidad del Enlace para paquetes recibidos
- Evaluación de Canal Libre para CSMA/CA
- Envío y Recepción de Datos

Como se mencionó anteriormente, cada capa posee un conjunto de servicios y entidades que se encargan de realizar las funciones asignadas.

La capa física incluye una unidad encargada de la gestión de los servicios de la capa, a través de la cual se invocan las funciones de

manejo y gestión. Dicha unidad también mantiene una base de datos de los objetos gestionados por la capa física, llamada PIB.

La capa física provee también dos servicios para la comunicación con la capa superior. El primero de los servicios es el Servicio de Datos, cuya función es dar soporte a los MPDU que vienen de la capa MAC. El otro servicio es el servicio de Gestión, servicio dedicado al transporte de los comandos de gestión entre el MLME y el PLME. Estos servicios son accedidos a través de dos SAP, el PD-SAP y el PLME-SAP.

2.2.5.3.2.1. Bandas de Frecuencia y Canales de Operación

El estándar ZigBee puede operar en 3 bandas de frecuencia distintas. Cada banda consta de un cierto número de canales de operación.

Las primeras dos bandas, 868 MHz y 915 MHz están reservadas para Europa y Estados Unidos únicamente. La tercer banda de frecuencias de 2450 MHz es una banda de frecuencias libre a nivel mundial (ISM en ingles o su traducción IMC: Industrial - Medica - Científica), por lo que se utiliza también en el resto del mundo.

El número de canales de operación depende de la banda de frecuencia, la primera banda contiene solo un canal, cuya frecuencia central es de 868.3 MHz. Para la segunda banda se disponen de 10 canales de operación y una separación de 2 MHz entre ellos. Y para la tercer banda de 2.4 GHz se cuenta con 16 canales de operación y una separación de 5 MHz. En la Figura N° 8 se puede apreciar el número de canales de cada banda de frecuencias.

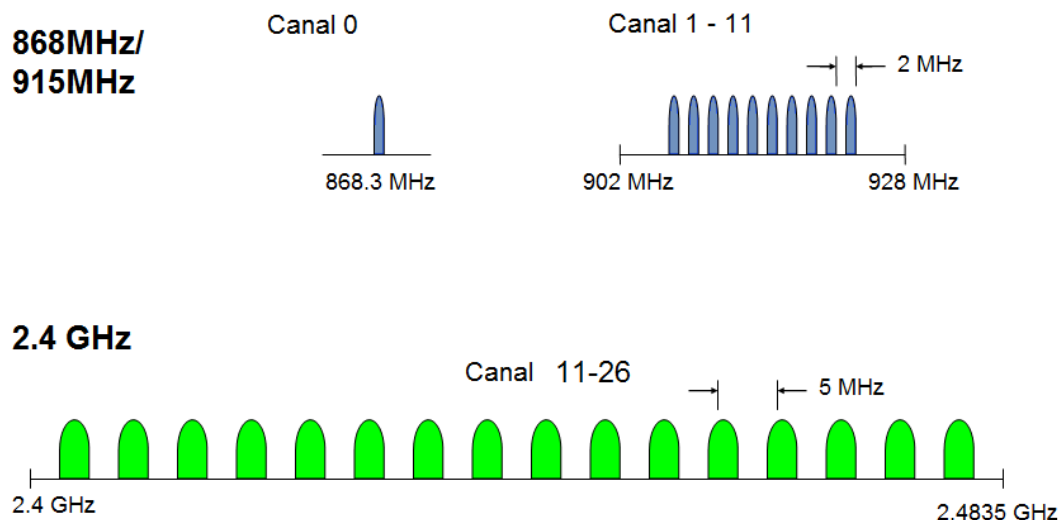


*Figura 8 – Canales de las Bandas de Frecuencia del estándar ZigBee
Imagen por Frederick Martin, Motorola Inc.*

2.2.5.3.2.2. Distancias de Coberturas

Las distancias de cobertura de los módulos IEEE 802.15.4 / ZigBee varían de acuerdo al entorno en el que se encuentran y al tipo de dispositivo. Los dispositivos ordinarios tienen un alcance de 40-50 metros en ambientes urbanos o de interiores, y de hasta 180 metros en exteriores. Los dispositivos de mayor potencia y costo, cuentan con alcances de hasta 100-120 metros en ambientes urbanos o interiores y de hasta 1,5 – 1,6 Kilómetros en exteriores.

2.2.5.3.3. Capa de Enlace

La capa de enlace está dividida en dos sub-capas: La de control de enlace lógico (Logical Link Control - LLC) y la sub-capa de acceso al medio (Medium Access Control - MAC).

LLC es una sub-capa común a todos los estándares 802, más precisamente es el 802.2 de tipo 1. Este accede a la MAC mediante dos SAP (Service Access Point):

- MCPS-SAP: Mac Common Part Sublayer - Utilizado para el enlace de la sub-capa MAC
- MLME-SAP: Mac Layer Management Entity. Sirve para los servicios de administración.

La sub-capa MAC 802.15.4 maneja todos los accesos al medio físico, y es responsable de tareas como:

- Generar las beacons, de ahora en adelante balizas, de la red si el dispositivo es coordinador
- Sincronizar las balizas
- Encargarse de la asociación y disociación de los dispositivos de la red.
- Brindar seguridad a la red
- Controlar el acceso al medio a través de CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance)

El estándar IEEE 802.15.4 implementa dos mecanismos de acceso al medio (MAC):

- Redes sin balizas: En este modo ZigBee utiliza CSMA/CA, para evitar las colisiones en el envío de datos. Se envían reconocimientos (Acknowledges) por cada paquete recibido.
- Redes con balizas: El coordinador de la red elige tiempos de balizamientos, los cuales pueden ser de entre 4 milisegundos y 4 minutos. Una vez elegido los tiempos de balizamiento, envía broadcast de balizas a la red. Los dispositivos finales pueden

estar en estado “sleep” durante varios de estos ciclos de balizas para ahorrar energía, luego despertar y esperar por la próxima baliza para saber si el coordinador tiene algún mensaje guardado para él. Si no hay mensajes, el dispositivo se vuelve a dormir.

En las comunicaciones inalámbricas, una baliza es una trama que envía el coordinador a la red para informar que este se encuentra activo y listo para transferir datos.

2.2.5.3.3.1. Asociación y Disociación

En el caso de la asociación de dispositivos, cada elemento final envía una solicitud que es recibida por el coordinador para registrar su presencia y formar parte de la PAN, esto permite incorporar un nuevo nodo a la red. Este proceso se denomina “descubrimiento de dispositivos”. Un nodo ZigBee puede unirse a la red, solo si se encuentra en el mismo canal de operación.

La disociación de un dispositivo se hace de forma análoga a la asociación, el coordinador de la PAN borra de su memoria la información y configuración del dispositivo a ser separado de la red. La disociación puede ser requerida por el coordinador o por el mismo módulo.

Si un módulo no puede ingresar a la red, el coordinador, mediante una transmisión indirecta o “pendiente” conoce el estado de conexión con el dispositivo final, y la solicitud de conexión queda en *stand by*.

Se debe destacar que, la asociación y disociación de dispositivos en una red con protocolo ZigBee (y no solamente IEEE 802.15.4), es

responsabilidad de la capa de Red, y no de la capa de enlace. Para este trabajo se utilizarán dispositivos con protocolo ZigBee, por lo que la asociación y disociación de dispositivos de la red, estará a cargo de la capa de Red, de la cual se habla más adelante.

2.2.5.3.3.2. Asignación de Direcciones

Los dispositivos ZigBee son asignados con direcciones de 64 bits y un direccionamiento opcional de 16 bits. Este último brinda mayor confiabilidad en redes de topología malla.

2.2.5.3.4. Capa de Red

La capa de red es la primera del estándar ZigBee propiamente dicho. Desarrollada por la *ZigBee Alliance*, su objetivo es proveer una correcta operación del estándar IEEE 802.15.4 a nivel MAC, así como también proporcionar una interfaz para los servicios de comunicación con la capa de aplicación.

También, por medio de la asignación de direcciones de red, permite que los dispositivos FFD sean utilizados como routers para poder ampliar los límites de la red. Esto permite topologías tipo mesh, como las descripta en el apartado 2.2.5.3.1.3

El servicio de datos de la capa de red, o NLDE por sus siglas en inglés (Network Layer Data Entity) provee el servicio de transmisión de datos a través de su SAP asociado, el NLDE-SAP. De forma similar el servicio de gestión de la capa de red, (NLME – Network Layer Management Entity), proporciona el servicio de gestión de la capa de Red por medio de su SAP correspondiente, el NLME-SAP. También es tarea del NLME mantener

una base de datos de los objetos gestionados, conocida como “Network Information Base – NIB” o base de información de la red, en español.

2.2.5.3.4.1. NLDE – Entidad de datos de la capa de Red

La NLDE debe proveer un servicio de datos que sea capaz de permitir a una aplicación el transporte de unidades de datos del protocolo de aplicación (APDU Application Protocol Data Units) entre dos o más dispositivos. Dichos dispositivos deben estar en la misma red.

Por PDU se denomina a la unidad de datos que genera cada sub-capa del estándar. La capa de red genera sus propias PDU, llamadas NPDU (Network sub-layer Protocol Data Unit)

La PDU se encarga de las siguientes tareas:

- Generación de NPDUs desde la PDU dentro de la sub-capa de red.
- Direccionamiento específico para cada topología: Debe poder transmitir una NPDU al dispositivo apropiado, ya sea el destinatario final del mensaje o un paso intermedio hacia el destinatario final en la cadena de comunicación.
- Seguridad: Poseer la habilidad para asegurar la confiabilidad e integridad de una transmisión.

2.2.5.3.4.2. NLME – Entidad de Gestión de la capa de Red

Encargada de la gestión de servicios de la capa de red, permite a una aplicación comunicarse con el resto del protocolo.

Debe encargarse de los siguientes servicios:

- Configuración de un nuevo dispositivo: Esta es la habilidad para configurar el protocolo para operar como es requerido. Opciones de configuración incluyen comenzar una operación como coordinador ZigBee o unirse a una red existente.
- Unirse y abandonar una red: Es la habilidad del dispositivo de unirse o abandonar una red por pedido propio, o por pedido del Coordinador o de un Router de la red.
- Asignación de Direcciones por parte del Coordinador o de los Routers de la red.
- Descubrir, reportar y recordar información de los vecinos de la red.
- Descubrir y recordar caminos de la red para permitir enviar los mensajes de forma más eficiente.
- Control de Recepción: Habilidad del dispositivo para controlar cuando el receptor está activado y por cuanto tiempo, permitiendo sincronización de la sub-capa MAC o por recepción directa.
- Enrutamiento o “Routing”: Es la habilidad para utilizar diferentes mecanismos de direccionamiento, como ser: unicast – broadcast – multicast para intercambiar mensajes de forma eficiente en la red.

2.2.6.Reconocimiento del Habla – RAH

El Reconocimiento del Habla (RAH) es una rama de la inteligencia Artificial. Es un proceso automatizado donde la entrada es el habla humana, y el sistema debe comprender lo que ha sido dicho. El RAH posee diversas aplicaciones como ser, programas de dictado (convertir el habla a texto), interfaces controladas por voz, motores de búsqueda, etc. [8]

Para que un sistema de RAH sea funcional se debe hacer cooperar un conjunto de informaciones que provienen de diversas fuentes. Estas informaciones son la acústica, la fonética, la léxica, para el decodificador acústico-fonético, el cual reconoce las palabras que se ingresaron en formato de audio al sistema, y las informaciones sintáctico-semánticas y pragmáticas para el modelo del lenguaje, encargado del análisis semántico y de reconocer el significado de la frase.

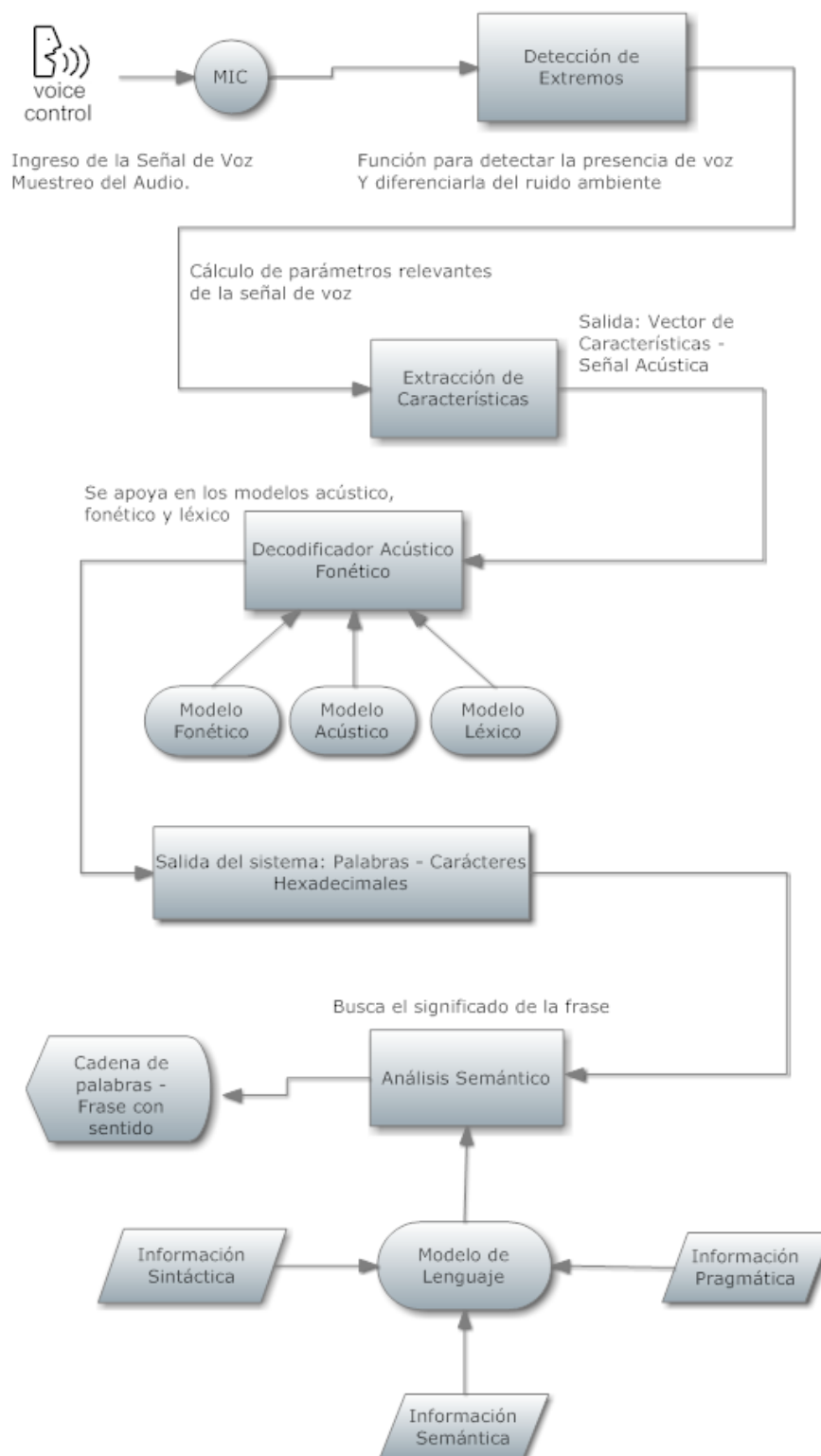


Figura 9 - Diagrama en bloques Sistema de Reconocimiento del Habla

Todo esto debe hacerse en la presencia de ciertas ambigüedades, de incertidumbres y de errores que son inevitables en cualquier persona que habla, para llegar a obtener una interpretación que sea manejable del mensaje acústico que fue recibido.

El reconocimiento automático del habla es una tecnología investigada desde mediados de la década del '50. Algunos laboratorios como Bell y RCA comenzaron desarrollando sistemas reconocedores de algunas sílabas pronunciadas por una misma persona

En los años '60 se establecen los primeros fundamentos en Japón, más precisamente en los laboratorios NEC. En la ex Unión Soviética, Dimitri Vintsyuk proponía la Alineación Dinámica de Tiempo (DTW – Dynamic Time Warping). Por otro lado, la Carnegie Mellon University (CMU) desarrolla el sistema de reconocimiento del habla continua llamado HAL 9000.

En la década del '70 se comienzan a dar las primeras soluciones gracias a la aplicación de las teorías probabilísticas. La IBM desarrolló proyectos de reconocimiento de grandes vocabularios. Estados Unidos hace una gran inversión en el proyecto DARPA. Además, la CMU finaliza el proyecto HARPY, primer sistema con éxito.

Para la década del '80 se habían logrado grandes avances en el reconocimiento del habla continua. Hay una explosión en el uso de los modelos estadísticos, como los Modelos Ocultos de Markov. También se Introducen otras técnicas de inteligencia artificial como las redes neuronales para el reconocimiento de voz y se desarrolla el sistema SPHINX.

Los años '90 comenzaron a comercializarse los primeros sistemas y aplicaciones para computadoras personales las cuales eran Sistemas de dictado para procesadores de texto, la integración de reconocimiento de voz y procesamiento del lenguaje natural.

La llegada del año 2000 vino acompañada de grandes avances en la aplicación del reconocimiento del habla mediante su integración a los sistemas operativos, teléfonos celulares, y otras aplicaciones.

Actualmente, el reconocimiento automático del habla tiene un gran desarrollo en la industria automovilística, de teléfonos celulares, del software y del hardware proporcionando soluciones para aplicaciones específicas. También, está comenzando a ser usado en la industria de la domótica, dando una comodidad extra o una respuesta a una necesidad de una persona con alguna discapacidad particular, como es el caso del presente trabajo.

Como ejemplos, puede citarse el sistema SYNC, desarrollado por Microsoft para Ford, e incorporado a los autos de su línea Kinetic, en la *figura 10* puede verse el sistema SYNC utilizado en un Ford Fiesta Kinetic. También el sistema de Toyota, Entune, que utiliza la red celular para enviar los comandos de voz, que son procesados en un servidor ubicado en la nube.



Figura 10 – SYNC en modo escucha funcionando en un Ford Fiesta Kinetic

Fuera de la industria del automotor, nos encontramos con Siri, herramienta de Apple capaz de reconocer una amplia gama de comandos, y de realizar búsquedas en la web, de lo ingresado por el usuario mediante su voz. Continuando en el rubro de los dispositivos móviles, encontramos S-Voice, de Samsung, con las mismas capacidades que el producto de Apple.

En la industria de la electrónica, HMC desarrollo el HM2007, capaz de reconocer hasta 40 palabras pre-grabadas por el usuario, de 1.92 segundos de duración cada una. Otro dispositivo disponible es el Easy VR, de Veeear, que puede reconocer 32 comandos de voz predefinidos, en 28 idiomas distintos. Puede encontrarse en su versión Shield para Arduino.

Por otro lado, Google desarrolló su sistema para el reconocimiento de voz basado en la nube, y lo incorporó a la API de su sistema operativo Android, por lo que cualquier desarrollador de aplicaciones, dentro del

universo Android, puede dotar a sus programas de esta poderosa herramienta.

2.2.6.1. Técnicas de Reconocimiento del Habla

Al comienzo del presente capítulo se mencionó que se debía interpretar un mensaje en medio de ambigüedades, errores en el habla, etc. Uno de los principales problemas, es que cuando una persona habla, nunca pronuncia dos palabras exactamente de la misma forma. Siempre hay una diferencia en la velocidad de pronunciación, por lo tanto, habrá una diferencia en el eje temporal de las secuencias de palabras.

Técnica	Característica
Alineamiento Temporal Dinámico	Alineación de la palabra a reconocer con una palabra de referencia. Se emplea Algoritmo para reconocer similitudes con la referencia.
Modelos Ocultos de Markov	Modelo estadístico para reconocer parámetros desconocidos a través de parámetros observables.
Redes Neuronales	Inteligencia Artificial. Red consistente en una serie de capas, que emula el comportamiento de una neurona real. Cada capa se implementa a través de un algoritmo llamado "backpropagation"

Tabla 1 – Técnicas de Reconocimiento del Habla más usadas

2.2.6.1.1. Alineamiento Temporal Dinámico (DTW)

El DTW es una de las técnicas más utilizadas al momento de diseñar un sistema de reconocimiento del habla. Su función es alinear la palabra a reconocer con la palabra de referencia, respecto del eje temporal para que el sistema luego pueda aplicar un método que le permita encontrar similitudes.

Las secuencias son alineadas de forma no lineal en un eje temporal. Luego de esto, se emplea un algoritmo distinto para buscar patrones y similitudes entre la palabra a reconocer y la palabra de referencia.

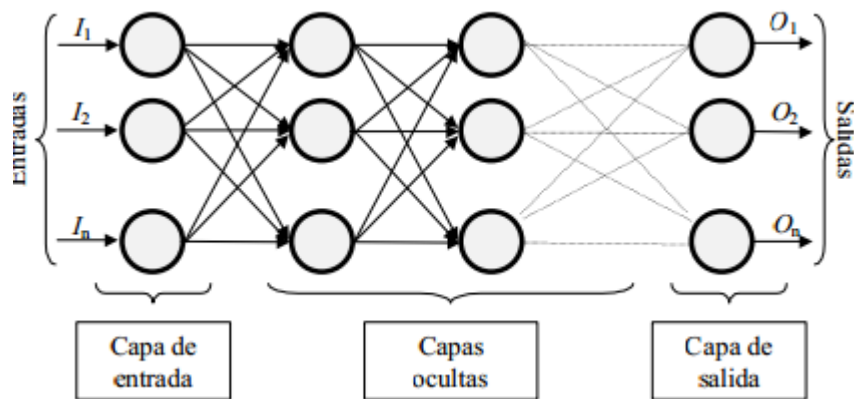
2.2.6.1.2. Modelos Ocultos de Markov (HMM)

Un Modelo Oculto de Markov es un modelo estadístico diseñado para el reconocimiento de parámetros desconocidos, a través de una serie de parámetros observables. Estos parámetros que son extraídos se pueden utilizar para llevar a cabo sucesivos análisis, un ejemplo de esto sería en aplicaciones de reconocimiento de patrones.

2.2.6.1.3. Redes Neuronales

Una red neuronal es un concepto de la Inteligencia Artificial que proporciona un modelo para imitar el funcionamiento de la neurona humana. Dicha red, consiste en una serie de capas, donde la salida de cada una de las capas es la entrada de la siguiente. Entre la capa de entrada y salida, puede haber un número finito de capas ocultas.

La implementación de estas redes se realiza computacionalmente a través de un algoritmo llamado “backpropagation” o retro propagación, por el cual se puede entrenar un sistema computarizado para que aprenda a reconocer ciertos patrones. [12]



[12]

Figura 11 – Diagrama de una Red Neuronal

Cuando las redes neuronales contienen un número finito de capas ocultas entre la capa de entrada y la de salida, se les llama Redes Neuronales Profundas. Cada unidad de cada capa oculta usa, típicamente, la función logística para mapear el total de las entradas que salieron desde la capa anterior y convertirlo al escalar que será enviado a la capa siguiente. [13]

La siguiente formula define la función logística:

$$y_j = \text{logistic}(x_j) = \frac{1}{1 + e^{-x_j}}, \quad x_j = b_j + \sum_i y_i w_{ij},$$

Dónde:

x_j es la salida de la capa anterior

y_j es el escalar enviado a la capa siguiente

b_j es la desviación de cada unidad oculta j

w_{ij} es el peso de la conexión entre una capa i y una capa j

Las redes neuronales han crecido considerablemente en los últimos años, gracias a la creciente velocidad y potencia de los nuevos procesadores, y del advenimiento de los procesadores gráficos o GPU

más orientados al cálculo de operaciones matemáticas, logrando ser cada vez más eficientes. Lo que permite construir redes neuronales con más capas de neuronas, y así poder resolver problemas más complejos.

2.2.7. Servicio de Google para el Reconocimiento de Voz.

En su última versión de Android, al momento de escribir este trabajo (Jelly Bean), Google incorporó un servicio renovado de reconocimiento automático del habla, que presenta una mejora considerable ante las versiones anteriores de dicho sistema operativo.

Esta poderosa herramienta integra la búsqueda web, con el control del dispositivo móvil en donde se encuentra instalado el S.O y permite servicios tales como dictado e ingreso de texto con una tasa muy baja de errores, búsqueda en la web, búsqueda en el sistema de mapas Google Maps, etc.

Se puede hacer uso de este servicio incorporándolo libremente a desarrollos y dotar así a las aplicaciones de una novedosa característica, como es el ingreso de órdenes mediante la voz humana.

2.2.7.1. Métodos para el RAH utilizados e implementados por Google

Como ya se mencionó anteriormente, existen diversas técnicas disponibles para el RAH, todas ellas en el campo de la Inteligencia Artificial y de la Probabilidad y Estadística.

En el caso del servicio de Google, el sistema está construido sobre una plataforma de procesamiento distribuida en los Data Centers que

posee la empresa en diversos puntos del planeta y una red neuronal que actúa de forma distribuida entre una serie de súper computadoras que se encuentran en dichos centros de datos.

Haciendo uso de este potente poder de cómputo, Google implemento una compleja red neuronal con miles capas ocultas, que le permiten obtener muy buenos resultados a la hora de entrenar los Modelos Ocultos de Markov utilizados para realizar el reconocimiento de los modelos lingüísticos.[13]

Un espectrograma de la voz del usuario es capturado por el dispositivo Android, y es enviado por internet hasta alguno de los Data Centers antes mencionado, donde es ingresado a la Red, interpretado y devuelto en formato de texto.[14]

2.3. Metodologías de Investigación

El tipo de estudio que corresponde a la investigación es de tipo descriptivo, ya que se pretende utilizar tecnologías ya estudiadas y aplicadas en otros campos similares. Y propositivo porque una vez finalizado se dará un producto terminado que solucione los problemas planteados por el proyecto.

2.3.1. Hipótesis

- La calidad de vida en el hogar de una persona con discapacidad visual, aumenta utilizando un sistema de domótica que le facilite las tareas cotidianas.
- Se mejora la seguridad personal en las tareas domésticas que le presentan mayor dificultad y riesgo, mediante la identificación y

automatización de las mismas.

2.3.2. Técnicas de Recolección de Datos

Con el fin de poder obtener un panorama más amplio y real sobre las necesidades de una persona con discapacidad visual dentro de su hogar, se decidió se decidió consultar a una persona con dicha discapacidad, a la cual se le consultó sobre cuáles son las necesidades más importantes a cubrir, en su diario manejar dentro del hogar.

3. DESCRIPCIÓN DEL SISTEMA DOMÓTICO

3.2 Descripción General del Sistema

El sistema se compone básicamente de un Nodo coordinador, un Acces Point con conectividad WiFi (de ahora en adelante “Router WiFi”), tres nodos finales (cantidad ampliable al número máximo que una red ZigBee permite), y un dispositivo Android.

La elección del Sistema Operativo Android se basó en la amplia disponibilidad de equipos que pueden correr dicho sistema y en su gran versatilidad. Como ejemplo de otros dispositivos diferentes a los Smartphones o tablets, pueden citarse las “Mini-PC”, que son pequeños dispositivos integrados que cuentan con gran cantidad de periféricos, conectividad LAN y buena capacidad de ejecución.



Figura 12 – Android Mini-PC

El nodo coordinador se encuentra conectado mediante una red Ethernet al router WiFi. El router WiFi en conjunto con el nodo coordinador, actúan a modo de Gateway entre la WSN e Internet. El coordinador recibe las instrucciones que el usuario ingresa mediante el cliente instalado en el dispositivo Android, Luego, el dispositivo Android toma las instrucciones de voz del usuario y las transforma en caracteres hexadecimales, que son enviados al nodo coordinador.

Los nodos se comunican entre ellos de forma inalámbrica utilizando los dispositivos XBee ZigBee.

El nodo coordinador, tomara los datos recibidos desde el cliente Android, interpreta las órdenes del usuario, y envía las ordenes necesarias a los nodos finales, que ejecutaran las acciones que el usuario había requerido, mediante los actuadores conectados a ellos, activando o desactivando luces, electroválvulas para control de fluidos y activando motores para abrir puertas o ventanas, a través de sus correspondientes etapas de potencia.

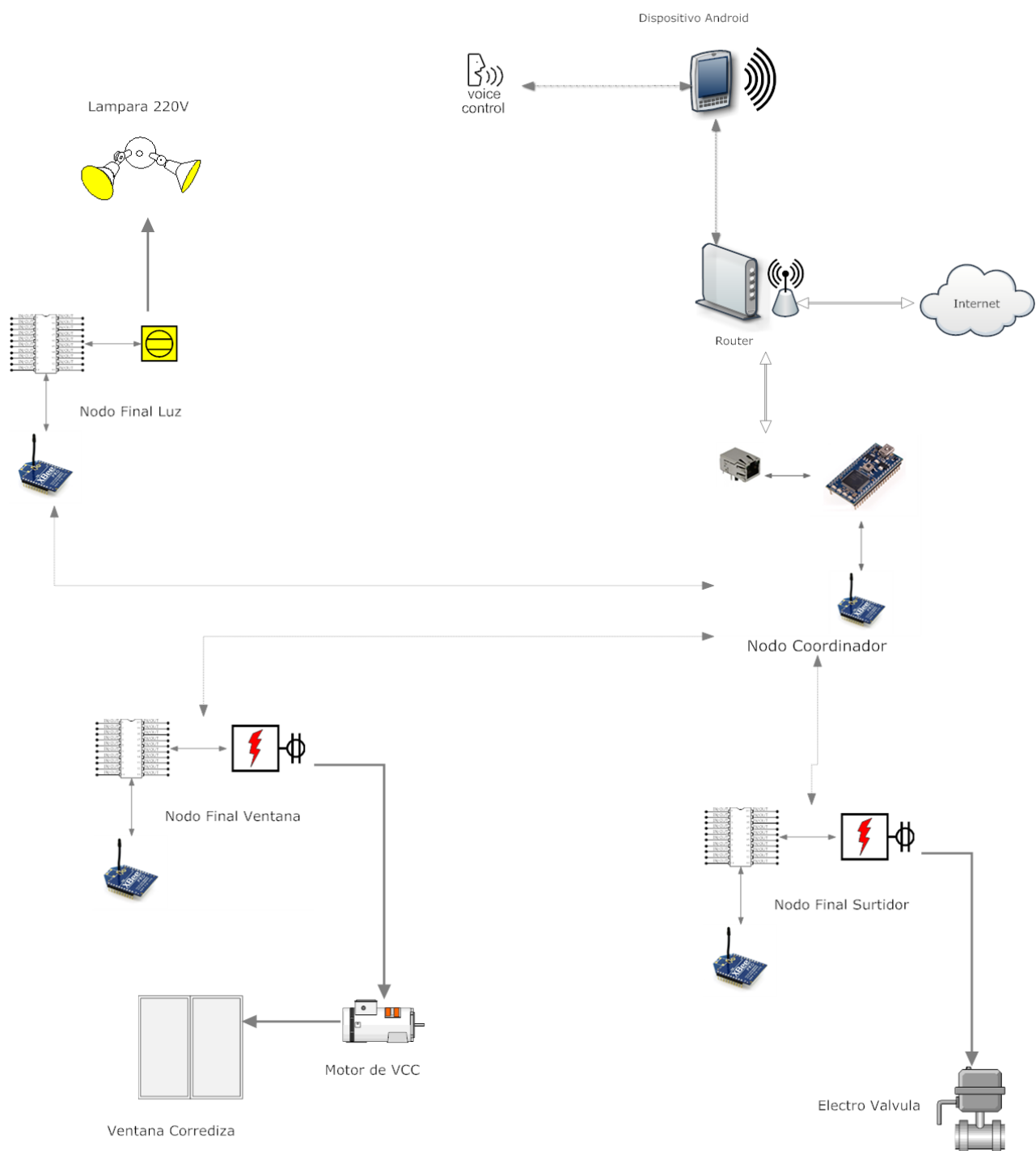


Figura 13 - Diagrama General del Sistema

3.3 Descripción del Hardware

El sistema está constituido por una WSN compuesta por los dispositivos mencionados número determinado de nodos. Estos nodos se encontrarán instalados en puntos estratégicos de la vivienda lo que permitirá el control de ciertas tareas de forma automática. Los nodos se comunicarán con un dispositivo coordinador de forma directa, o a través de un router ZigBee en el caso de que nodo y coordinador no puedan comunicarse directamente debido a distancia u obstáculos.

3.3.1 Coordinador

Este es el nodo principal y gestor del sistema domótico. En él está incluido el microcontrolador principal marca ARM modelo LPC1768 de NXP, embebido en un dispositivo Mbed, un transceptor ZigBee FFD y un puerto Ethernet para dotar al sistema de comunicación Web. Mediante un Router con acceso a internet o una red WiFi, el coordinador se comunica con un dispositivo Android, que es el encargado de tomar las órdenes de voz del usuario. En la *figura 14* se muestra un diagrama de bloques del nodo coordinador

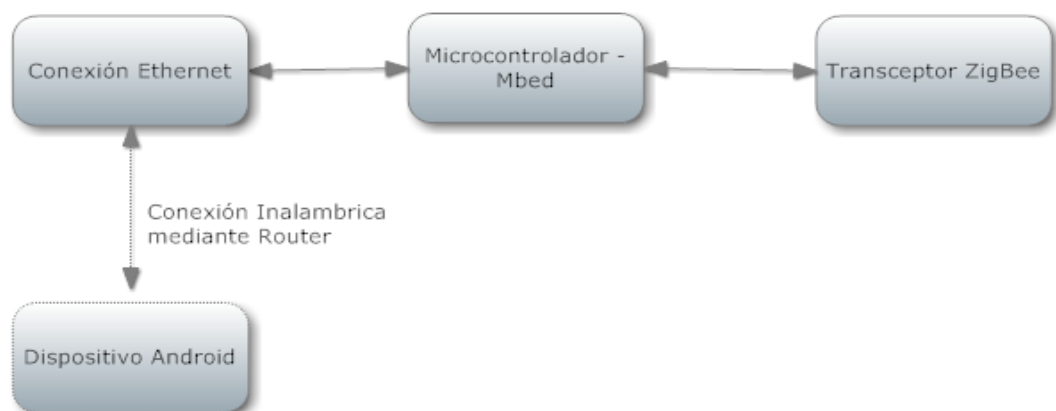


Figura 14 - Diagrama en Bloques del Coordinador

El microcontrolador Mbed tiene la posibilidad de ser alimentado de dos formas distintas. Una es mediante el puerto Mini-USB que posee para programación, Debug y comunicación con la PC. La segunda forma posible, es mediante alimentación por fuente o batería, de entre 4.5V y 9V, directo a la entrada de alimentación del mbed, que está regulada a 3.3V.

La alimentación del resto del circuito, se efectúa mediante una salida regulada del dispositivo Mbed, a 3.3V y cuya capacidad de entrega de corriente es suficiente para satisfacer las necesidades del módulo ZigBee en su etapa de transmisión

La conexión entre el transceptor ZigBee y el Mbed se realiza mediante los puertos series de ambos dispositivos y no se utiliza control de flujo, pero se encuentra preparado por si fuese necesario.

El Mbed cuenta con el hardware necesario para conectarse a una red Ethernet. El único requisito es la adaptación a los niveles de tensión necesarios, esto fue realizado mediante un RJ45 marca "MagJack" con las bobinas transformadoras necesarias.



Figura 15– Mbed LPC1768

Elemento	Cantidad	Características
Procesador	96 Mhz	ARM Cortex-M3 Core
RAM	32 KB RAM	
Memoria	512KB Flash	
Ethernet	1	10-100 Mbps
USB	1	Host/Device
SPI	2	
I2C	2	
UART	3	
CAN	1	
PWM	6	
ADC	6	
Alimentación 1	5V	USB
Alimentación 2	4.5V – 9V	Fuente regulada
Programación		USB Drag N' Drop
Lenguaje del compilador	2	C o C++

Tabla 2 - Características Técnicas MBED

3.3.2 Nodo Final

El nodo final es el encargado de sensar las variables de entorno y ejecutar las tareas de automatismo mediante los actuadores. Estará compuesto de uno o más sensores o actuadores o ambos, dependiendo de la tarea de la cual esté encargado, de un microcontrolador el cual tomara decisiones locales respecto de las lecturas que indican los sensores y ordenará a los actuadores realizar una acción correctiva. También tendrá un transceptor ZigBee mediante el cual se comunicará con el nodo Coordinador y por este con el resto de la red.

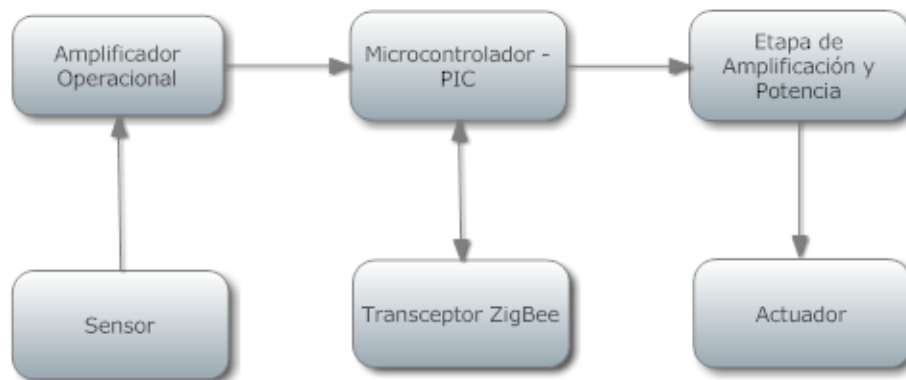


Figura 16 - Diagrama en Bloques del Nodo Final

Constructivamente, el nodo final tendrá una etapa de amplificación por cada sensor con el que cuente, y una etapa de amplificación y potencia para el manejo de cada uno de los actuadores que posea.

La *figura 16* muestra su correspondiente diagrama de bloques.

La alimentación de cada Nodo puede ser mediante una batería de 9V o mediante una fuente de alimentación conectada a la instalación del hogar, mediante un transformador. La tensión mínima que requiere es de 4.5V y la tensión máxima que tolera es de 30V. Cada nodo está regulado a 3.3V.

La comunicación entre el Microcontrolador y el transceptor ZigBee es a través de sus puertos serie, y el manejo de los actuadores es mediante las entradas y salidas del Microcontrolador.

Las etapas de Potencia y Amplificación están separadas físicamente de las placas de los nodos finales por cuestiones de seguridad y para evitar posibles interferencias

3.4 Software para Reconocimiento de los comandos de Voz

Para resolver el reconocimiento de voz del sistema, se desarrolló una aplicación utilizando el lenguaje Python [15] y la API de Android de Google [16], que provee, aparte de los servicios ya descritos en el punto 2.2.7 del marco teórico escrito en este trabajo, un entorno para desarrollar en lenguaje Java, así como también pueden desarrollarse aplicaciones en Python, Perl, etc.

Dicha aplicación, que corre en cualquier dispositivo con sistema Android, es un cliente web conectado al servidor que se ejecuta en el coordinador. El cliente, utiliza la función “recognize speech” para tomar las órdenes de voz que el usuario ingresa, y guarda el texto devuelto en una variable de tipo cadena.

Luego, el cliente envía al servidor en el nodo coordinador los comandos de voz ya procesados. Por último, se cierra la conexión y se queda a la espera de que se ingrese un nuevo comando de voz. Puede verse un diagrama de flujos de la aplicación en la siguiente imagen.

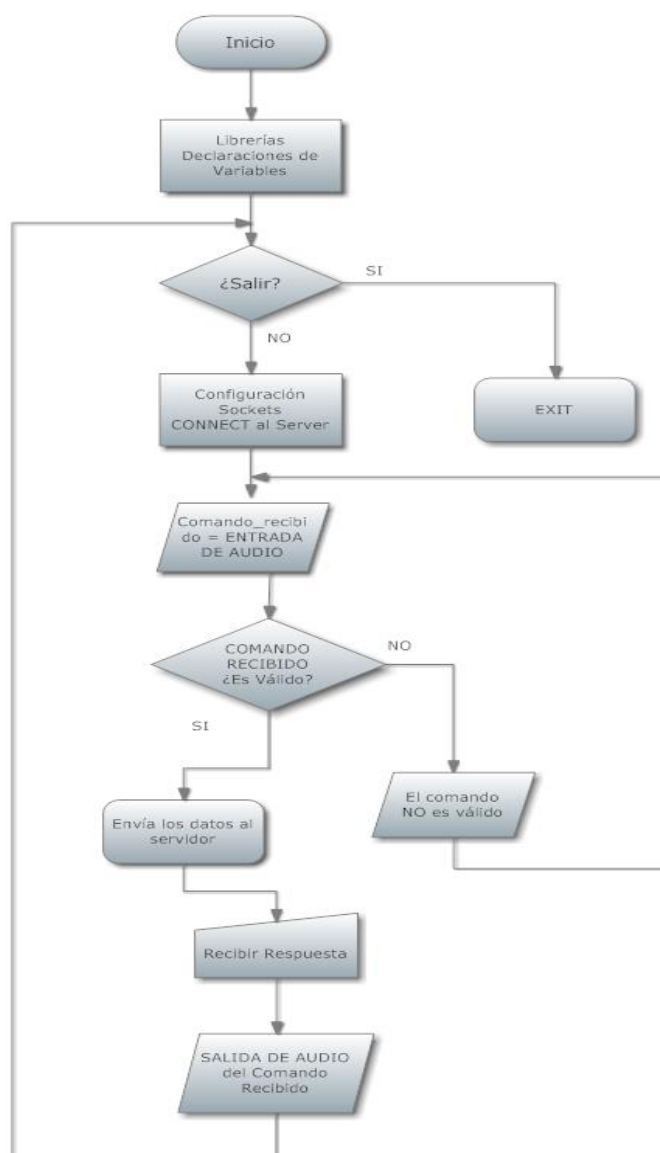


Figura 17 – Diagrama de Flujo del Cliente Android

La ventaja de este enfoque tipo cliente web, en un dispositivo Android, es que por su versatilidad, puede accederse al sistema a través de Internet, conllevando esto a manejar las aplicaciones, o vigilar y controlar las variables, desde cualquier lugar del globo en el cual se tenga acceso a la red.

3.5 Descripción del Firmware

El Hardware es controlado mediante un Firmware. Cada nodo tendrá un Firmware específico para la tarea que le toca realizar. Así, el Firmware del coordinador general será el más complejo, y los Firmware de los nodos finales de la red tendrán una complejidad dependiente de la función o funciones que tendrán asignadas.

3.5.1 Firmware Coordinador

El Firmware del dispositivo Coordinador es el encargado de controlar las acciones que reciba desde el cliente de reconocimiento de voz, y los parámetros generales del sistema. También es el encargado de enviar mensajes con nuevos parámetros u órdenes a los nodos finales, o recibir de ellos mensajes con nuevos datos medidos en los sensores.

El Firmware está diseñado con un formato de servidor web, el cual es el responsable de escuchar de forma permanente los comandos del usuario que vienen desde el cliente, el ciclo de funcionamiento se completa con un módulo encargado de construir las tramas de comunicaciones, que serán enviadas por la red ZigBee a los nodos finales con las órdenes para cada nodo, según corresponda.

En la *figura 18* se puede observar un diagrama de flujo del funcionamiento general del Firmware del coordinador, en donde se detalla el proceso que lleva a cabo.

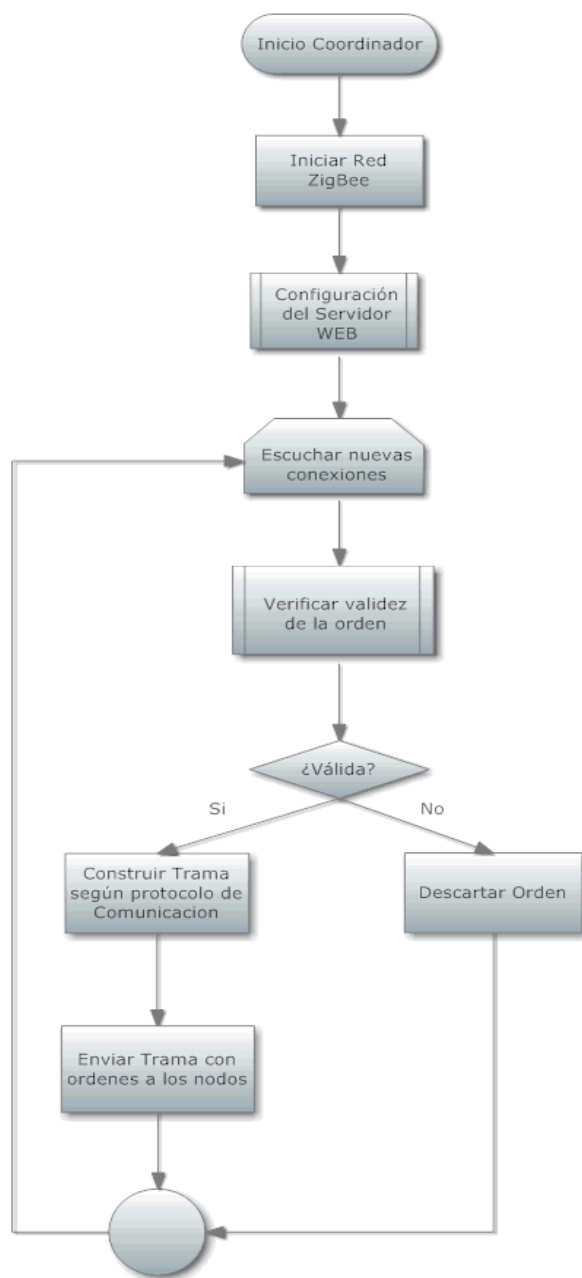


Figura 18- Diagrama de Flujos del Firmware del Coordinador

3.5.2 Firmware Nodo Final

El Firmware del nodo final controla cada tarea de automatización. También maneja los tiempos de sensado del entorno. Se muestra su diagrama de flujo en la *figura 19*.

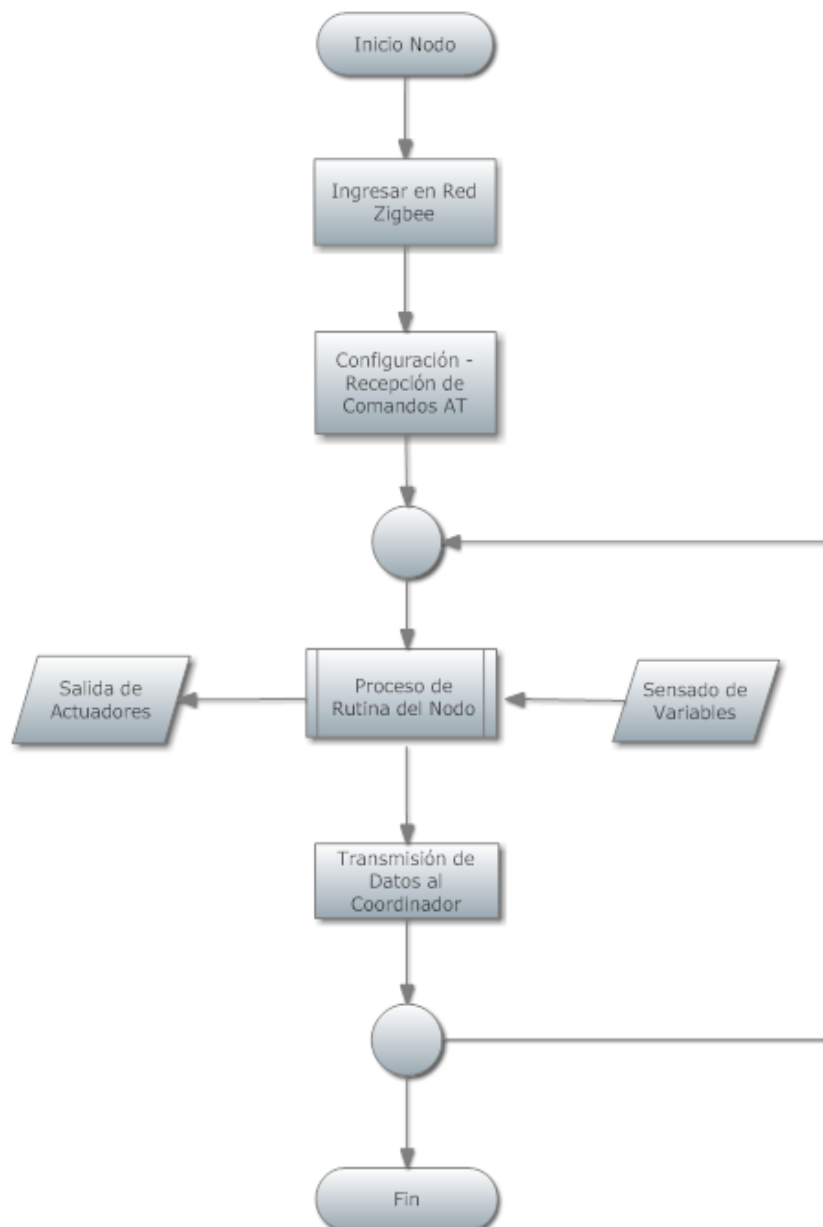


Figura 19 - Diagrama de Flujo del Firmware del Nodo Final

El microcontrolador está escuchando permanentemente la entrada de alguna trama por su puerto serie, proveniente desde el módulo ZigBee. Para determinar si una trama es válida, la descompone y revisa que su cabecera, el primer byte de la misma, sea la correcta. Si es válida, pasa a leer el segundo byte, que indica el largo de la trama. Los 2 siguientes campos de bytes, contienen la orden y el objeto que deben operar. Toda esta tarea es realizada por una función, ejecutada cada 1ms mediante la operación del Timer del microcontrolador.

Según la orden recibida, se colocan en “uno” una serie de flags que accionaran las entradas o salidas digitales, las cuales a su vez activarán las etapas de potencia de cada objeto del hogar que desea ser utilizado.

Por último, se envía al coordinador un acuse de recibo para que este sepa que la trama ha sido recibida de forma exitosa.

En el caso de detectar bytes inválidos o una trama inválida, la misma se desecha y no se envía un acuse de recibo, lo que provocará que el coordinador re-envíe la trama.

4. DESARROLLO DE INGENIERÍA

En el presente capítulo se describe el proceso de desarrollo y construcción del sistema, así como las dificultades que se sortearon durante el mismo.

4.1. Desarrollo del Hardware

4.1.1. Hardware del Nodo Coordinador

Debido a la gran cantidad de hardware incorporado que cuenta un sistema embebido como es el Mbed LPC1768, se necesitan adicionar pocos componentes para comenzar a desarrollar un prototipo funcional.

El circuito impreso se diseñó para evitar las posibles interferencias electromagnéticas y para protegerlo frente a señales de ruido radioeléctrico, derivadas del uso del módulo de radiofrecuencia, que utiliza señales de 2.4GHz. Siguiendo esta línea de diseño, se rellenaron todos los espacios posibles cercanos al módulo con planos de masa.

Otro concepto de diseño, fue posicionar el módulo de comunicaciones XBee, de tal modo que su antena quede fuera del límite de la placa impresa, para reducir la posibilidad de inducciones en las líneas de transmisión de datos.

Para energizar el circuito, se utiliza la salida regulada del Mbed, de 3.3V y 800mA, según la hoja de datos su regulador, el LT1117S33 de Linear Technology. El Mbed puede ser alimentado a través de su puerto USB o mediante su entrada desde una fuente de 4.5V - 9V.

En cuanto a la comunicación inalámbrica, se utilizó un módulo XBee-PRO Serie 2 (ZB), cuyas características técnicas se aprecian en la tabla 3. Se colocó un capacitor de 100uF en la entrada de alimentación del módulo XBee como filtro ante posibles componentes de alta frecuencia, que pudiesen haber sido inducidas en la línea.

ESPECIFICACIONES		XBeePRO-ZB
Rendimiento	Velocidad de Datos en RF	250 Kbps
	Alcance en interiores/Zonas Urbanas	90m
	Alcance en exteriores/Línea de visión	1,6Km
	Potencia de Transmisión	50mW (17dbm)
	Sensibilidad de Recepción	-102 dbm
Información General	Interfaz Serie	LV-TTL (3.3V)
	Configuración	AT y API
	Banda de Frecuencias	2.4GHz
	Velocidad Serie	1200bps–1Mbps
Red y Seguridad	Encriptación	AES 128 bits
	Entrega confiable de paquetes	Reenvíos/ACK
	Identificadores y Canales	64bits IEEE MAC 13 Canales
	Rango de Alimentación	3 – 3.4 Vcc
Propiedades Eléctricas	Corriente en Transmisión	295mA @ 3.3Vcc
	Corriente en Recepción	45mA @ 3.3Vcc
	Corriente en Bajo Consumo	<10uA @ 25°C

Tabla 3 – Características Técnicas XBee PRO-Serie2

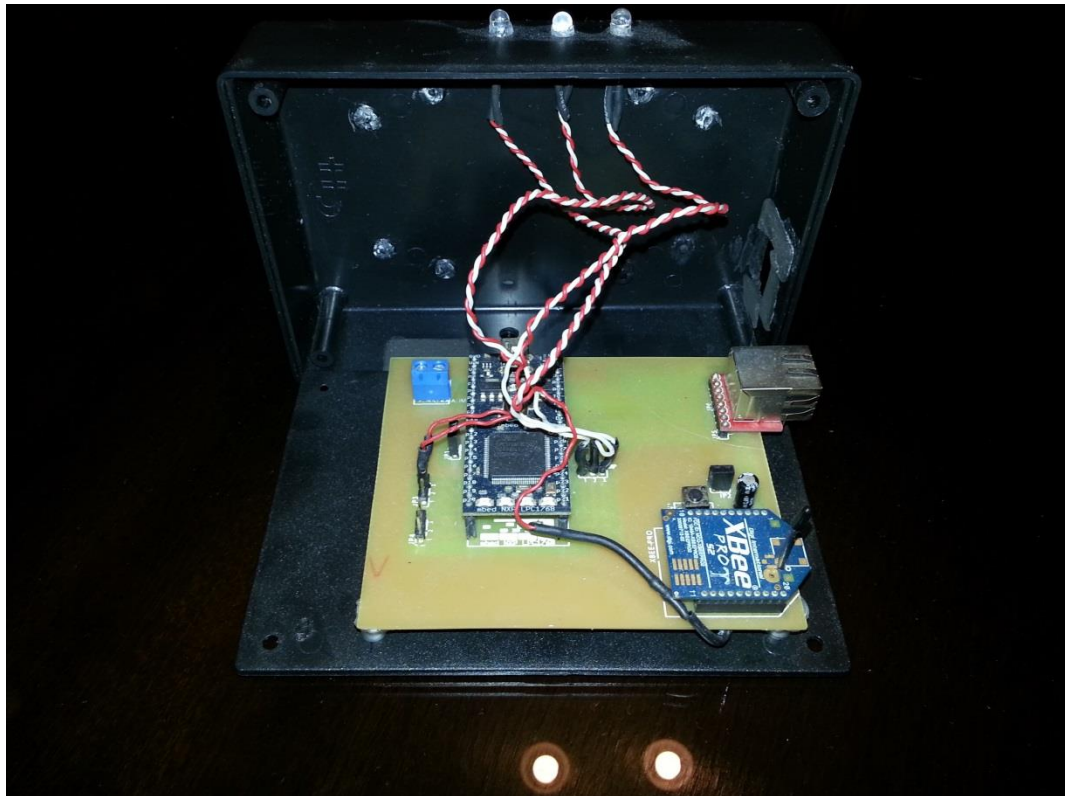


Figura 20 – Nodo Coordinador

4.1.2. Hardware del Nodo Final

Los nodos finales fueron construidos en base a un único diseño, con leves modificaciones según la aplicación a la que son destinados.

Las mismas consideraciones, en cuanto al diseño del circuito impreso que se tuvieron para el diseño del coordinador, se mantuvieron para el diseño de los nodos finales, con el objetivo de proteger al circuito de posibles interferencias producidas por el módulo de radiofrecuencia.

El microcontrolador que se eligió fue el PIC16F876-A, con un cristal que le permite funcionar a 10Mhz. La alimentación del circuito es de 3.3V,

provistos por un LM317, regulador que puede soportar hasta 30V de tensión continua con un límite de trabajo de hasta 1.5 A de corriente.

El módulo de comunicaciones es un XBee Serie 2 (ZB), y sus especificaciones técnicas se ven en la tabla 4.

ESPECIFICACIONES		XBee ZB
Rendimiento	Velocidad de Datos en RF	250 Kbps
	Alcance en interiores/Zonas Urbanas*	40m
	Alcance en exteriores/Línea de visión*	120m
	Potencia de Transmisión	2mW (3dbm)
	Sensibilidad de Recepción	-96dbm
Información General	Interfaz Serie	LV-TTL (3.3V)
	Configuración	AT y API
	Banda de Frecuencias	2.4GHz
	Velocidad Serie	1200bps–1Mbps
Red y Seguridad	Encriptación	AES 128 bits
	Entrega confiable de paquetes	Reenvíos/ACK
	Identificadores y Canales	64bits IEEE MAC 16 Canales
	Rango de Alimentación	2.1 – 3.6Vcc
Propiedades Eléctricas	Corriente en Transmisión	45mA @ 3.3Vcc
	Corriente en Recepción	40mA @ 3.3Vcc
	Corriente en Bajo Consumo	<1uA @ 25°C

Tabla 4 – Características Técnicas XBee Serie 2

** El alcance logrado depende de factores ajenos a la potencia de transmisión y sensibilidad de recepción tales como atenuación por espacio libre, ganancia de las antenas, objetos en línea de visión y zonas de Fresnel.*

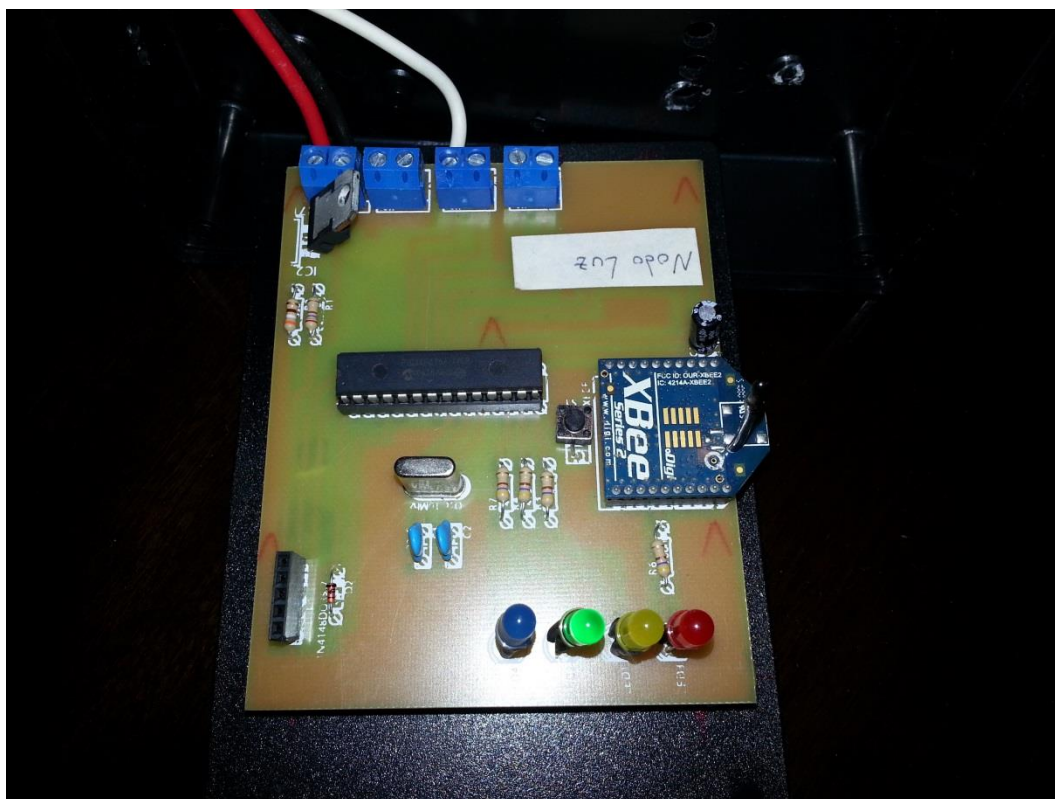


Figura 21 – Nodo Final

4.2. Desarrollo del Firmware

4.2.1. Desarrollo del Firmware del Nodo Coordinador

El proceso de diseño y desarrollo del firmware del nodo coordinador se basó en la necesidad de contar con un dispositivo que permitiera recibir vía red Ethernet, internet o LAN, las instrucciones que deben ser enviadas a los nodos finales cuya tecnología de comunicación es el protocolo ZigBee.

El Mbed cuenta con un RTOS (Real Time Operative System) escrito en su librería, que permite implementar hilos que se ejecutan de forma paralela al bucle principal del programa. Se hizo uso de esta capacidad

para la implementación de una función que actualiza una serie de contadores internos con fines genéricos para la ejecución del programa.

Para permitir la escalabilidad del Coordinador, el firmware se diseñó de tal manera que pudiesen agregarse funciones de forma modular, sin perder su practicidad y utilidad como servidor. Inclusive, el uso de hilos provee capacidad para escalar el sistema de forma sencilla, dando características de ejecución en tiempo real, como una máquina de estados, independiente su ejecución de los eventos que se sucedan, y otorgándole robustez y confiabilidad al nodo coordinador.

Las funciones principales son 3:

- Servidor: Maneja la configuración TCP/IP, estando permanentemente escuchando el puerto de entrada por nuevas conexiones del cliente y provee la información recibida al siguiente módulo del programa.
- Manejador de Protocolo: Funcionalmente es el cerebro del sistema domótico, responsable de clasificar las órdenes y construir la trama adecuada para la orden recibida.

```
40 //**** Funcion que arma la trama de comunicaciones ****//
41 void constr_proto(char* header, char* c_com, char* c_obj, char* c_id, char* tram, int length)
42 {
43     int l_head = 0, l_com = 0, l_obj = 0, l_id = 0, l_trama = 0;
44     char ln_trama[2];
45     l_head = strlen(header);
46     l_com = strlen(c_com);
47     l_obj = strlen(c_obj);
48     l_id = strlen(c_id);
49     l_trama = l_head + l_com + l_obj + l_id + 1;
50     ln_trama[0] = (char)l_trama;
51     ln_trama[1] = 0x00;
52     printf("%d\n\r", l_trama);
53     strncat (tram, header, l_head);
54     strncat (tram, ln_trama, strlen(ln_trama));
55     strncat (tram, c_id, l_id);
56     strncat (tram, c_com, l_com);
57     strncat (tram, c_obj, l_obj);
58 }
59
```

Figura 22 – Manejador de protocolo

- Manejador de Puerto Serie: Encargado de la comunicación vía puerto serie con el XBee.

```
60 //**** Funcion para enviar trama al Modulo Xbee ****//
61 void enviar_xbee(int length, char* trama_tx){
62
63     int i = 0;
64
65     for (i = 0; i < length; i++){
66         xbee1.putc(trama_tx[i]);
67     }
68
69     for (i = 0; i < length; i++){
70         trama_tx[i] = 0;
71     }
72
73 }
```

Figura 23 – Función de envío al módulo XBee

4.2.2. Desarrollo del Firmware del Nodo Final

La necesidad de garantizar la fluidez, rapidez y confiabilidad del sistema domótico, lleva a que cada eslabón del mismo debe contar con estas tres mencionadas características.

Cada nodo final tiene un microcontrolador PIC, que debe responder a los requerimientos del sistema de forma rápida y confiable, a pesar de su limitada capacidad de procesamiento, por lo que el diseño de su firmware fue optimizado para responder rápidamente siempre, aún ante cualquier eventualidad.

Con los objetivos anteriormente planteados, es que se optó por diseñar el firmware en forma de máquina de estados, otorgándole un tiempo de procesamiento determinado a cada función del programa, y

estas últimas trabajando como cajas negras, donde la entrada de una función es la salida de otra. Esta característica también permite la escalabilidad y la modificación del firmware de cada nodo final, para que con pequeños cambios en su codificación puedan atender el control de distintas aplicaciones.

El firmware de cada nodo cuenta con un bucle principal y una serie de funciones que se ejecutan cada un cierto periodo de tiempo.

- Lector de Protocolo: Encargado de interpretar los caracteres que entran por puerto serie y recibe la función de comunicación entrante. Verifica que dichos caracteres sean de una trama de comunicaciones válida y libre de errores. También controla que su nodo sea el destinatario de la trama recibida.

```
void leer_prot()
{
    if ((cad_rx[0] == 0x7E) && (cad_rx[2] == ID_NODO)) //Busca cabecera
    {
        if (cad_rx[3] == 0x41 && cad_rx[4] == 0x42 && cad_rx[5] == 0x52) //Comando Abrir
        {
            if (cad_rx[8] == 0x41 && cad_rx[9] == 0x47 && cad_rx[10] == 0x55) //Objeto: Agua
            {
                opt = 0x10;
                borra_cadena();
            }
            borra_cadena();
        }

        if (cad_rx[3] == 0x43 && cad_rx[4] == 0x45 && cad_rx[5] == 0x52) //Comando Cerrar
        {
            if (cad_rx[9] == 0x41 && cad_rx[10] == 0x47 && cad_rx[11] == 0x55) //Objeto: Agua
            {
                opt = 0x20;
                borra_cadena();
            }
            borra_cadena();
        }

        borra_cadena();
    }
    else
    {
        borra_cadena();
    }
}
```

Figura 24 – Función leer_prot del nodo final

En la *Figura 24* puede verse la implementación de esta función, en la primera sentencia *if* busca la cabecera y que la trama haya sido enviada con destino a su nodo. Luego, en las siguientes sentencias *if* se buscan los caracteres hexadecimales correspondientes a la trama válida de la aplicación que debe manejar su nodo final, en este caso de abrir o cerrar la electroválvula para permitir el paso del agua.

- Interrupción de puerto serie: Recibe los caracteres que entran al puerto serie desde el XBee de su nodo, y los guarda en un arreglo, el cual es la entrada del lector de protocolos.
- En el bucle principal se ejecutan las acciones que el lector de protocolo encontró válidas, y activa o desactiva las entradas o salidas digitales del PIC, según corresponda activar o desactivar la aplicación que controla.

4.3. Descripción Funcional del Sistema

4.3.1. Funcionamiento

Ante la ya explicada finalidad del presente sistema domótico, el diseño fue concebido para que implique un mínimo costo en esfuerzo de parte del usuario final.

Con solo un comando de voz, el usuario puede, encender o apagar luces, abrir o cerrar la llave de agua corriente, ya sea para higiene, consumo o riego. También abrir o cerrar ventanas, puertas o cortinas.

Para realizar el set-up del sistema, se deben seguir los siguientes pasos:

- 1) Conectar el Coordinador a la alimentación y a la red LAN del hogar, obteniendo su IP fija. El coordinador creará una Red ZigBee de forma automática.
- 2) Instalar y conectar a la alimentación los Nodos Finales. Estos se unirán a la red ZigBee automáticamente.
- 3) En un dispositivo Android conectado a la red LAN, abrir la aplicación de control por voz. La primera vez que se realiza esta acción se deberá ingresar la IP del coordinador al que se desea conectar el cliente.
- 4) Emitir una orden válida de voz. El sistema actuará según la orden recibida.
- 5) Si se desea salir de la Aplicación, se deberá pronunciar el comando SALIR.

Puede verse un diagrama de bloques del funcionamiento del sistema según la vista del usuario final en la *figura 25*. Los procesos dentro del recuadro azul, son invisibles para el usuario final.

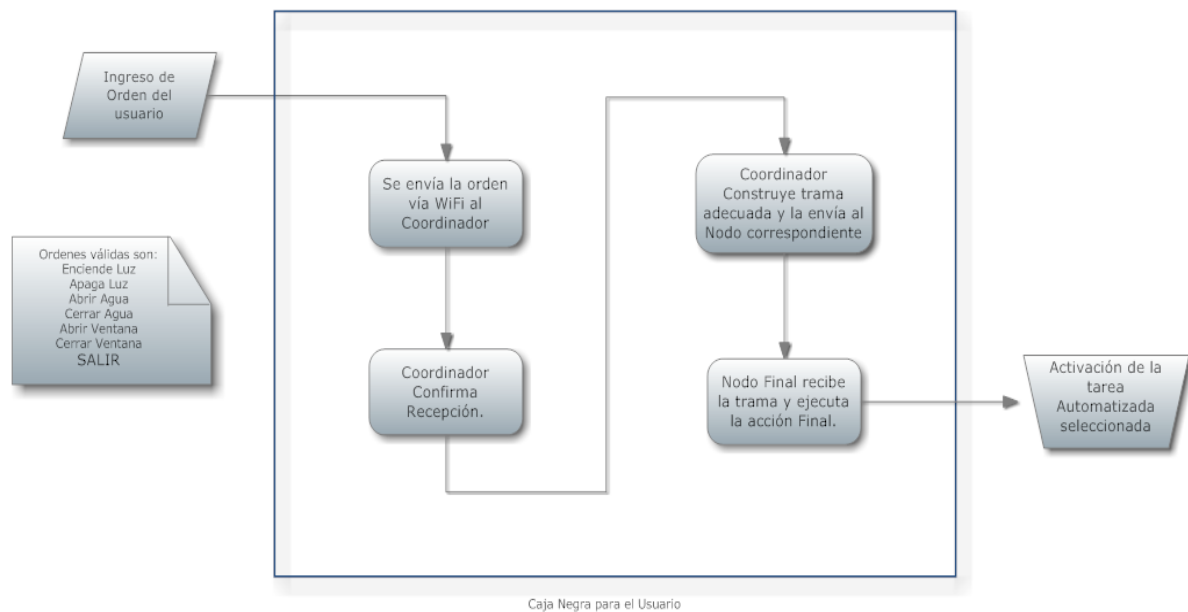


Figura 25 – Diagrama de funcionamiento según vista del usuario final

5. ANALISIS DE COSTOS

En el presente capítulo se detallan los costos de cada parte del sistema. Dichos costos están expresados en Dólares y en Pesos Argentinos.

Valor dólar BCRA U\$S1 = AR\$8.03 – Impuestos considerados: IVA 10.5% Importaciones – Gravamen sobre compras con crédito en moneda extranjera 35% (Recuperable a fin de año)

Nodo Coordinador

<i>Detalle</i>	<i>Cantidad</i>	<i>Precio</i>		<i>Total en AR\$</i>
		U\$S	AR\$*	
Dispositivo Mbed	1	49.95	598.33	598.33
MagJack RJ45	1	1.95	23.35	23.35
XBee Pro Serie 2	1	40.95	527.06	527.06
Circuito Impreso	1	-	65	65
Componentes Varios	-	-	30	30
Gabinete	1	-	50	50
Total				1293.74

Nodo Final

<i>Detalle</i>	<i>Cantidad</i>	<i>Precio</i>		<i>Total en AR\$</i>
		U\$S	AR\$*	
PIC 16F876A	1	5.44	65.13	65.13
LM 317	1	-	12	12
XBee Serie 2	1	25.95	310.84	310.84
Circuito Impreso	1	-	65	65
Componentes Varios	-	-	60	60
Gabinete	1	-	50	50
Total		----		562.97

Instalaciones y Cableados

<i>Detalle</i>	<i>Cantidad</i>	<i>Precio AR\$</i>	<i>Total AR\$</i>
Cable Rojo 2A	3 m	1.23	3.69
Cable Blanco 2A	1 m	1.23	1.23
Cable Amarillo 2A	1 m	1.23	1.23
Cable Negro 2A	3 m	1.23	3.69
Termo contraíble	1 m	5.76	5.76
Total	----		15.6

Etapas de Potencia Nodo Ventana

<i>Detalle</i>	<i>Cantidad</i>	<i>Precio</i>		<i>Total en AR\$</i>
		U\$S	AR\$	
LM293B	1	-	32	32
7805	1	-	5.27	5.27
Bornera	4	-	3.29	13.16
Circuito Impreso	1	-	35.5	35.5
Componentes Varios	-	-	10	10
Pilares	4	-	5	20
Gabinete	1	-	35	35
Total		----		150.93

Etapas de Potencia Nodo Luz

<i>Detalle</i>	<i>Cantidad</i>	<i>Precio</i>		<i>Total en AR\$</i>
		U\$S	AR\$	
Transistor BD 139	1	-	3.75	3.75
7812	1	-	5.27	5.27
Bornera	3	-	3.29	9.87
Circuito Impreso	1	-	35.5	35.5
Relay	1	-	18.3	18.3
Pilares	4	-	5	20
Gabinete	1	-	50	50
Total		----		142.69

Etapas de Potencia Nodo Válvula

<i>Detalle</i>	<i>Cantidad</i>	<i>Precio</i>		<i>Total en AR\$</i>
		U\$S	AR\$	
Transistor BD 139	1	-	3.75	3.75
7805	1	-	5.27	5.27
Bornera	4	-	3.29	13.08
Circuito Impreso	1	-	20.5	20.5
Pilares	4	-	5	20
Gabinete	1	-	50	50
Total		----		112.68

Extras

<i>Detalle</i>	<i>Cantidad</i>	<i>Precio</i>		<i>Total en AR\$</i>
		U\$S	AR\$	
Dispositivo Android*	1	38.99	467.05	467.05
Fuente 12V – 500mA	3	-	100.65	301.95
Electro válvula 12V	1	20	239.57	239.57
Total				1008.57

**El dispositivo Android puede no ser necesario adquirirlo si se cuenta con un smart-phone o tablet con S.O. Android de Google.*

Total

<i>Detalle</i>	<i>Cantidad</i>	<i>Precio</i>		<i>Total en AR\$</i>
		U\$S	AR\$	
Nodo Coordinador	1	-	1293.74	1293.74
Nodo Final	3	-	562.97	1688.91
Instalaciones y Cableados	1	-	15.6	15.6
Etapas Potencia Electro Válvula	1	-	112.68	112.68
Etapas Potencia Luz	1	-	142.69	142.69
Etapas Potencia Ventana	1	-	150.93	150.93
Extra	1		1008.57	1008.57
Total		----		4413.12

6. CONCLUSIONES

Como primer punto, se destaca el crecimiento y la experiencia en el ámbito académico, la experiencia que dejó haber encarado un proyecto desde su inicio, y la investigación sobre el estado actual de los temas teóricos.

En el ámbito personal, se logra el objetivo de desarrollar un prototipo funcional de un sistema domótico, concebido y desarrollado para aportar mejoras en la calidad de vida de personas cuya motricidad se encuentre disminuida debido a diferentes causas.

Respecto al desarrollo del sistema, se lograron comandar tareas automatizadas mediante órdenes de voz que se imparten a un teléfono celular, permitiendo que personas con discapacidad motriz o visual, puedan activar o desactivar ciertas tareas previamente automatizadas de su hogar sin la ayuda de terceras personas, o con una asistencia mínima de ellas.

Se construyó un sistema que no requiere el cableado de grandes extensiones, por lo que el tiempo para su instalación queda reducido a un mínimo.

Dados los puntos anteriormente mencionados, se concluye que, el sistema cumple el objetivo de mejorar la calidad de vida de un usuario con discapacidad motriz o visual.

Por otro lado, se puede concluir que los dispositivos ZigBee, en particular los módulos XBee de Digi, de fácil implementación y gran potencia de uso, relativo bajo costo, y bajo consumo de corriente, son una

excelente alternativa al momento de proveer a un sistema, de comunicación inalámbrica.

Como resultado concluyente con respecto al dispositivo de reconocimiento del habla, se puede afirmar que la tecnología de Google provista a partir de su API, que fue aplicada en este proyecto, cumple con el objetivo de capturar órdenes de voz, y transformarlas en información útil y comprensible para el sistema domótico. Como contrapunto, depende de una conexión permanente a la red para su funcionamiento, lo que lo hace dependiente de ésta.

Finalmente se desean enumerar algunas conclusiones generales sobre lo que dejó este trabajo en el autor, para su experiencia, tanto académica como personal, y el esfuerzo que conlleva haber realizado un primer trabajo de investigación donde se construyó un prototipo para demostrar el funcionamiento de su aplicación.

BIBLIOGRAFÍA

Referencias

[1] “Villa Domótica”: Vivienda a escala automatizada con Simatic S7-200. Felipe Mateos Martín; María Reyes Poo Argüelles; Marta García Prado; Rosana Olaiz García. Universidad de Oviedo.

[2] Enciclopedia Online Wikipedia: <http://es.wikipedia.org/wiki/Sensor>

[3] ZigBee Alliance - <http://www.zigbee.org>

[4] P. Godoy; L. Iacono; O. Marianetti; C. García Garino. “Aplicaciones de Redes de Sensores Inalámbricos al sensado de variables ambientales y agrícolas - Estudio de los efectos de la potencia de transmisión en la vida útil y en la confiabilidad del enlace.” Segundo Congreso Virtual Los Microcontroladores y sus Aplicaciones. 2010

[5] Jin-Shyan Lee, Yu-Wei Su y Chung-Chou Shen: *“A Comparative Study of Wireless Protocols: Bluetooth, UWB, ZigBee, and Wi-Fi”*

[6] J. M. Giachino, “Smart sensors,” *Sensors and actuators*, 10(1986) 239-248.

[7] Vijay Raghunathan, Curt Schurgers, Sung Park, and Mani B. Srivastava. “Energy Aware Wireless Sensor Networks”. Department of Electrical Engineering, University of California, Los Angeles, CA 90095.

[8] Mikko Kurimo, Kalle Palomaki, Vesa Siivola, y otros. "Automatic speech recognition". Reporte Bianual 2006-2007. Universidad Tecnológica de Helsinki. Capítulo 8.

[9] Pere Ponsa "Domótica accesible: tecnologías al alcance de personas con discapacidad". Luis A. Torres. Doctorado en Control, Visión y Robótica. 4all-L@b. GREC. CETpD

[10]Xavier Caballero. "La Domotización al servicio de las personas con gran discapacidad física". Lola de la Fuente. ESCLAT. ESCLATEC.

[11] Hoja de Datos Disponible en:
www.digi.com/support/kbase/kbaseresultdetl.jsp?kb=125

[12] Damián Jorge Matich "Redes Neuronales: Conceptos Básicos y Aplicaciones." Universidad Tecnológica Nacional – Facultad Regional Rosario. Marzo de 2001.

[13] Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara Sainath, and Brian Kingsbury. "Deep Neural Networks for Acoustic Modeling in Speech Recognition". Signal Processing Magazine. 2012.

[14] Johan Schalkwyk, Doug Beeferman, Francoise Beaufays, Bill Byrne, Ciprian Chelba, Mike Cohen, Maryam Garret, Brian Strobe. "Google Search by Voice: A case study". Google, Inc. Advances in Speech Recognition: Mobile Environments, Call Centers and Clinics, Springer (2010), pag 61-90.

[15] <http://www.python.org>

[16] <http://developer.android.com/intl/es/guide/index.html>

Bibliografía Impresa

Abuchaem, Jamil. "El Proceso Diagnostico en el Adulto, el Niño y el Adolescente. Concepto de entrevista pautada". 1979. Ediciones Kargieman. Buenos Aires.

Caprile, Sergio R. "XBee y XBee-PRO ZB: ZigBee PRO". Comentario Técnico CTC-058. 2009. Cika Electrónica.

Institute of Electrical and Electronics Engineers. Especificación del estándar IEEE 802.15.4-2003. Año de Publicación 2004
<http://standards.ieee.org/getieee802/download/802.15.4-2003.pdf>

Lipovski G. Jack. "Introduction to microcontrollers". Segunda Edición. 2004. El sevier. Inglaterra.

Muhamad Ali Mazidi, Rolin D. McKinlay. "Pic Microcontroller and Embedded Systems".2008. Prentice Hall.

Pete Cross. "Zeroing in on ZigBee (Part 1) Introduction to the Standard"
En: Circuit Cellar. N° 165, Febrero 2005.
<http://www.circuitcellar.com/library/print/0205/Cross175/index.htm>

Ramón Pallás Areny. "Sensores y Acondicionadores de Señal". Cuarta edición. 2005. Marcombo.

Siquier de Ocampo, García Arzeno, Grassano. "Las técnicas proyectivas y el proceso psicodiagnostico". Cap. II "La entrevista Inicial". Ediciones Nueva visión. Buenos Aires.

Dayashankar Dubey "Smart Sensor". M.Tech. credit seminar report, Electronic Systems Group, EE Dep., IIT Bombay. Noviembre 2002

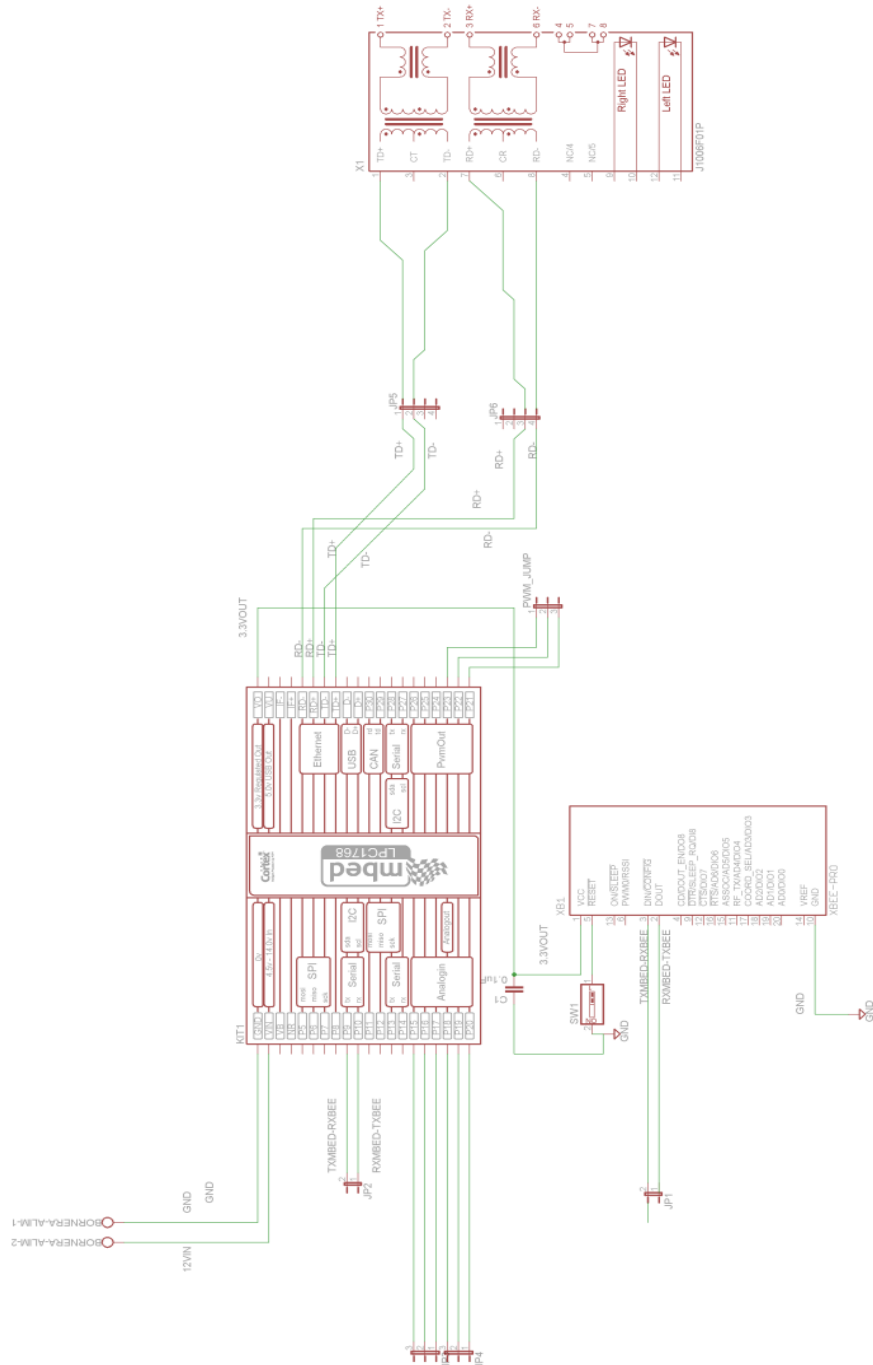
ZigBee Alliance. Especificación del estándar ZigBee. Año de publicación 2007.http://www.zigbee.org/zigbee/en/spec_download/spec_download.asp?AccessCode=990528257

ZigBee Alliance. Home Automation Profile. Año de publicación 2007.
http://www.zigbee.org/zigbee/en/spec_download/spec_download.asp?AccessCode=1018276353

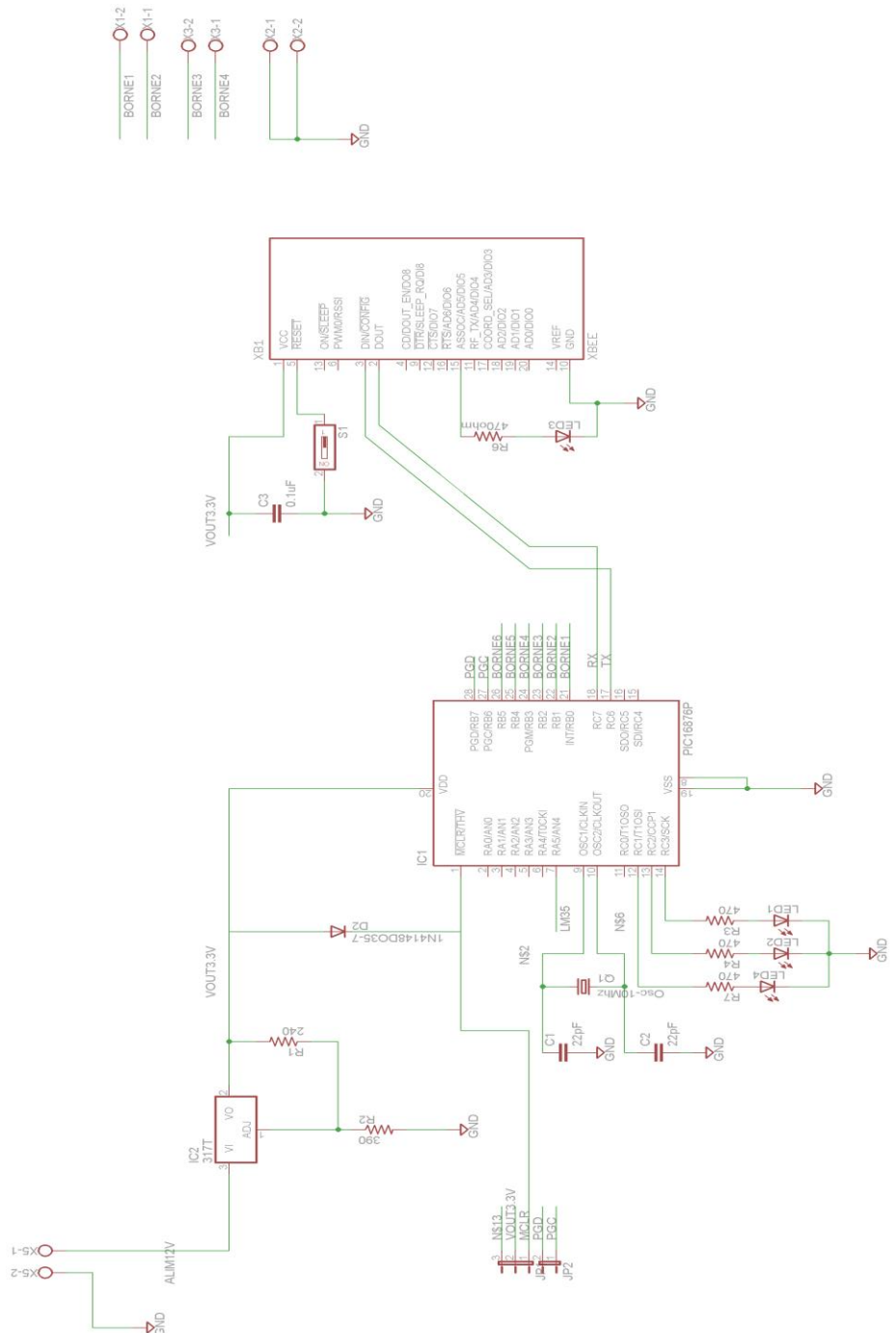
ANEXOS

ANEXO A – ESQUEMATICOS

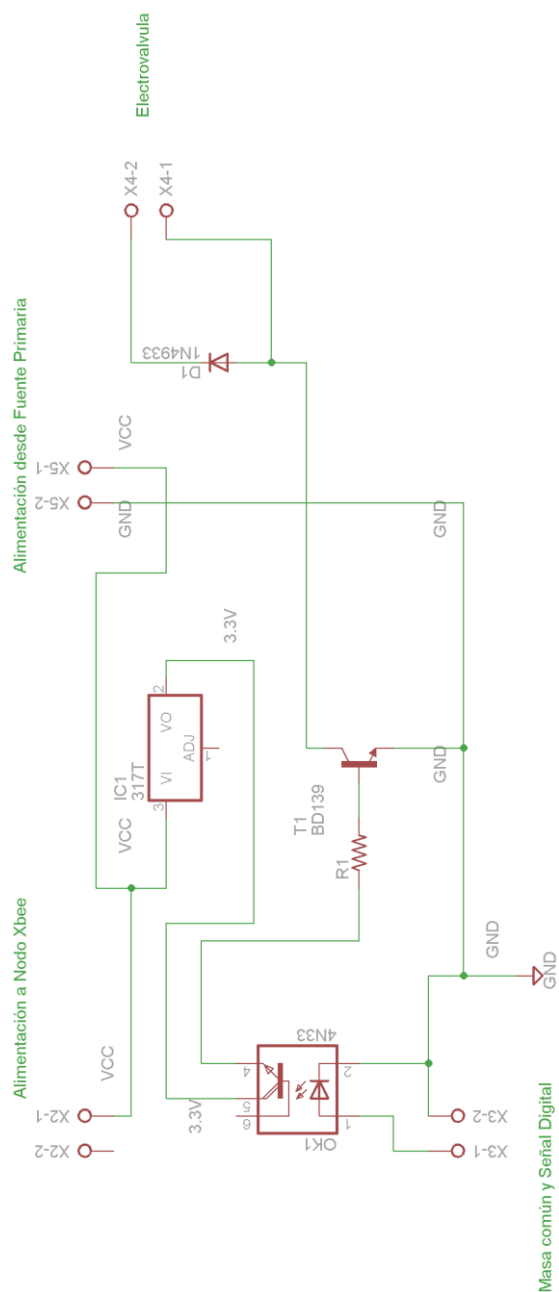
Nodo Coordinador



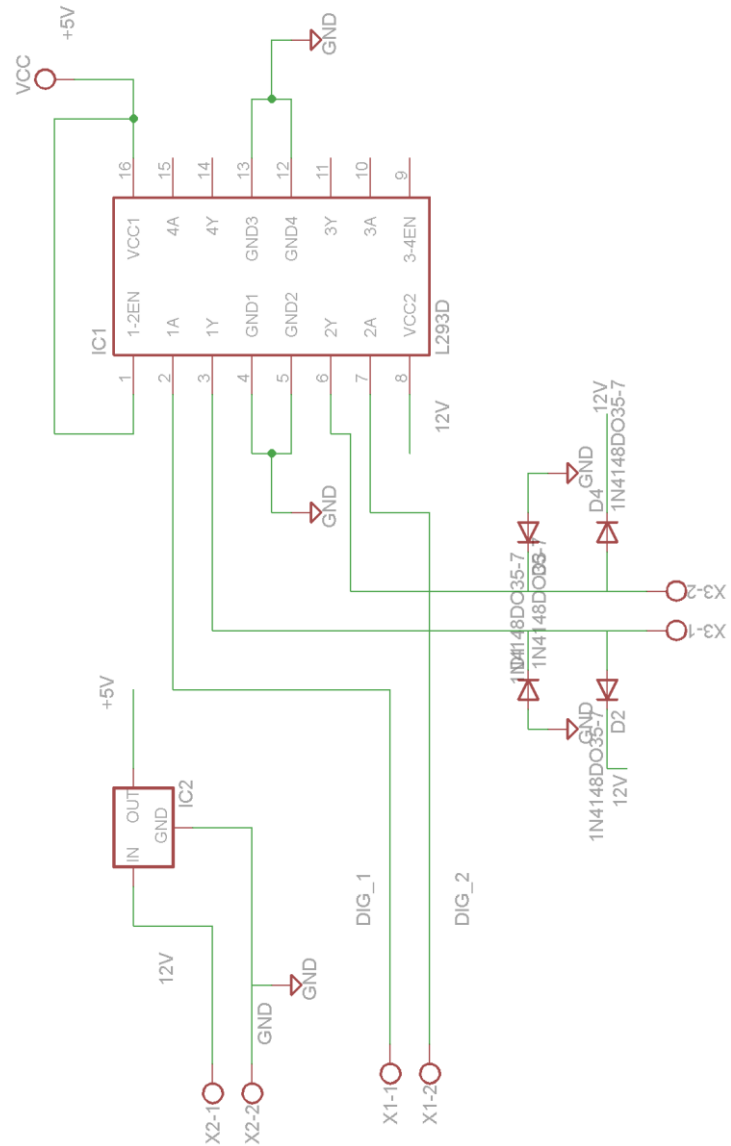
Nodo Final



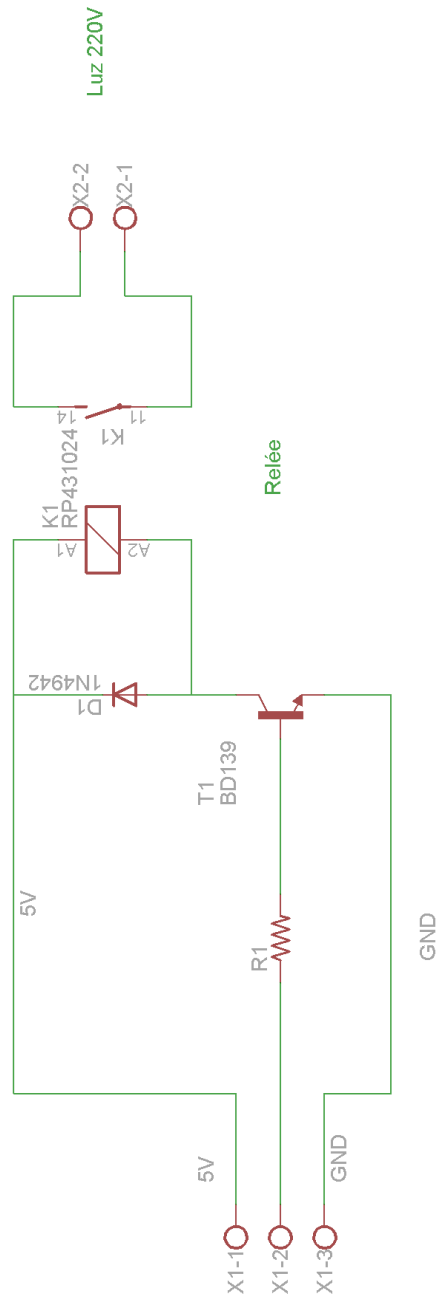
Circuito Potencia Válvula



Circuito Potencia Motor Ventana



Circuito Potencia Luz



ANEXO B – CÓDIGO FIRMWARE

Firmware Coordinador

```
#include "mbed.h"
#include "tramas.h"
#include "EthernetInterface.h"
#include <stdio.h>
#include <string.h>

#define PORT 7778
EthernetInterface eth;
Serial xbee1(p9, p10);
Serial usb(USBTX, USBRX);

TCPSocketServer svr;
bool serverIsListened = false;

TCPSocketConnection client;
bool clientIsConnected = false;

DigitalOut led1(LED1); //Estado de escucha del server
DigitalOut led2(LED2); //Estado de conexion del server
DigitalOut led3(LED3);
DigitalOut led_ext(p15); //Estado de escucha del server
DigitalOut led_ext_2(p16);
Ticker ledTick;
//*** Funcion que titila un led ***//
void ledTickfunc()
{
    if(serverIsListened) {
        led1 = !led1;
        led_ext = !led_ext;
    } else {
        led1 = false;
        led_ext = false;
    }
}

//**** Funcion que arma la trama de comunicaciones ****//
void constr_proto(char* header, char* c_com, char* c_obj, char* c_id, char* tram, int length)
{
    int l_head = 0, l_com = 0, l_obj = 0, l_id = 0, l_trama = 0;
    char ln_trama[2];
    l_head = strlen(header);
    l_com = strlen(c_com);
    l_obj = strlen(c_obj);
    l_id = strlen(c_id);
    l_trama = l_head + l_com + l_obj + l_id + 1;
    ln_trama[0] = (char)l_trama;
    ln_trama[1] = 0x00;
    printf("%d\n\r", l_trama);
    strncat (tram, header, l_head);
```

```

        //usb.printf("%s\n\r", tram);
        strncat (tram, ln_trama, strlen(ln_trama));
        //usb.printf("%s\n\r", tram);
        strncat (tram, c_id, l_id);
        //usb.printf("%s\n\r", tram);
        strncat (tram, c_com, l_com);
        //usb.printf("%s\n\r", tram);
        strncat (tram, c_obj, l_obj);
        // usb.printf("%s\n\r", tram);
    }

    /**** Funcion para enviar trama al Modulo Xbee *****/
    void enviar_xbee(int length, char* trama_tx)
    {
        int i = 0;

        for (i = 0; i < length; i++) {
            xbee1.putc(trama_tx[i]);
        }

        for (i = 0; i < length; i++) {
            trama_tx[i] = 0;
        }
    }

    /**** Funcion Principal *****/
    int main (void)
    {
        ledTick.attach(&ledTickfunc,0.5);           //Interrupción para llamar a
        la función cada 0.5 segundos

        // Variables Auxiliares
        int leng = 0;

        //Setup la interfaz de ethernet
        eth.init(); //Usar DHCP
        eth.connect();
        usb.printf("Direccion IP es %s\n\r", eth.getIPAddress());

        //Setup TCP socket
        if(svr.bind(PORT)< 0) {
            usb.printf("TCP Server - Bind Failed - Enlace Fallado.\n\r");
            return -1;
        } else {
            usb.printf("TCP Server - Bind Succeded - Enlace Correcto.\n\r");
            serverIsListened = true;
        }
        if(svr.listen(1) < 0) {
            usb.printf("TCP Server - Listen Failed - Escucha Fallada.\n\r");
            return -1;
        } else {
            usb.printf("TCP server esta escuchando...\n\r");
        }

        //Escucha nuevas conexiones..

        while (serverIsListened) {

```

```

//Modo Bloqueante - loop infinito (never timeout)
if(svr.accept(client)<0) {
    usb.printf("Falla al aceptar la conexion.\n\r");
} else {
usb.printf("Conexion exitosa\n\rIP: %s\n\r",client.get_address());
clientIsConnected = true;
led2 = true;
led_ext_2 = true;

while(clientIsConnected) {
    char buffer[128] = {};
    switch(client.receive(buffer, 127)) {

        case 0:
            usb.printf("Buffer recibido esta vacio.\n\r");
            clientIsConnected = false;
            break;

        case -1:
            usb.printf("Falla al leer datos del cliente.\n\r");
            clientIsConnected = false;
            break;

        default:
            //***** Logica de la aplicacion *****//

            //***** Comando ENCIENDE *****//
            if(buffer[0] == 'e' && buffer[1] == 'n' && buffer[2] == 'c') {

                //***** Enciende la Luz *****//
                if (buffer[9] == 'l' && buffer[10] == 'u' && buffer[11] == 'z')
                {
                    constr_proto( head, c_enc, c_luz, id_nodo_luz, trama,leng );
                    //Construyo la trama para encender la Luz
                    leng = strlen(trama);
                    enviar_xbee(leng, trama);    //Envio la trama al Modulo XBee
                    usb.printf("Comando Enciende Luz enviado.\n\r");
                    client.send("OK",2);
                    clientIsConnected = false;
                    leng = 0;
                }
            }

            //***** Comando ABRIR *****//
            if(buffer[0] == 'a' && buffer[1] == 'b' && buffer[2] == 'r') {
                usb.printf("Comando abrir.\n\r");
                client.send("OK",2);
                clientIsConnected = false;
                //***** Abrir el Agua *****//
                if (buffer[6] == 'a' && buffer[7] == 'g' && buffer[8] == 'u')
                {

                    constr_proto( head, c_abrir, c_agua, id_nodo_valvula, trama, leng);
                    //Construyo la trama para abrir el Grifo de Agua
                    leng = strlen(trama);
                    enviar_xbee(leng, trama); //Envio la trama al Modulo de comunicacion
                    usb.printf("Respuesta enviada.\n\r");
                    leng = 0;
                }

                //***** Abrir la Ventana *****//
                if (buffer[6] == 'v' && buffer[7] == 'e' && buffer[8] == 'n')

```

```

{
constr_proto( head, c_abrir, c_ventana,id_nodo_ventana, trama, leng);
//Construyo la trama para abrir la ventana
    leng = strlen(trama);
    enviar_xbee(leng, trama); //Envio la trama al Modulo de comunicacion
    usb.printf("Respuesta enviada.\n\r");
    leng = 0;
}
} //Cierro comando Abrir

        //***** Comando APAGAR *****//
if(buffer[0] == 'a' && buffer[1] == 'p' && buffer[2] == 'a') {
usb.printf("Comando apagar.\n\r");
client.send("OK",2);
clientIsConnected = false;
        //***** Apaga la Luz *****//
if (buffer[7] == 'l' && buffer[8] == 'u' && buffer[9] == 'z')
{
    constr_proto( head, c_apa, c_luz,id_nodo_luz ,trama, leng);
//Construyo la trama p/ apagar la luz (falta agregar checksum)
    leng = strlen(trama);
    enviar_xbee(leng, trama); //Envio la trama al modulo Xbee
    usb.printf("Respuesta enviada.\n\r");
    leng = 0;
}
}

        //***** Comando CERRAR *****//
if(buffer[0] == 'c' && buffer[1] == 'e' && buffer[2] == 'r') {
usb.printf("Comando cerrar.\n\r");
client.send("OK",2);
clientIsConnected = false;
        //***** Cerrar el Agua *****//
if (buffer[7] == 'a' && buffer[8] == 'g' && buffer[9] == 'u')
{
    constr_proto( head, c_cerrar, c_agua,id_nodo_valvula , trama, leng);
    leng = strlen(trama);
    enviar_xbee(leng, trama); //Envio la trama al Módulo de comunicación
    usb.printf("Respuesta enviada.\n\r");
    leng = 0;
}

        //***** Cerrar la Ventana *****//
if (buffer[7] == 'v' && buffer[8] == 'e' && buffer[9] == 'n')
{
    constr_proto( head, c_cerrar, c_ventana,id_nodo_ventana , trama, leng);
//Construyo la trama para cerrar la ventana (falta agregar checksum)
    leng = strlen(trama);
    enviar_xbee(leng, trama); //Envio la trama al Módulo de comunicación
    usb.printf("Respuesta enviada.\n\r");
    leng = 0;

}
}

        break;
}
}
usb.printf("Conexion cerrada.\n\rTCP server esta escuchando...\n\r");
client.close();
led2 = false;
led_ext_2 = false;
}
}

```

}

Firmware Nodo Electro Válvula

```
//*****//
/*
/*          NODO CONTROLADOR ELECTRO VALVULA          */
/*          Autor: Guillermo Paladini                  */
/*          */
//*****//

#include "F:\Guille\Electronica - Computacion\Proyectos\Nodo_Xbee\main.h"
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define 0x2000 + FLASH_ID_NODO = {ID_NODO}

int opt = 0, i = 0;
char cad_rx[32], c_rx;
long contrTCC = 0;
long contTAREA = 0;

void comm_in(void);
void borra_cadena(void);
void led_tick(void);
void leer_prot(void);

void led_tick()
{
    if(contrTCC <= 150)
    {
        output_high(PIN_B1);
    }
    if (contrTCC > 150)
    {
        output_low(PIN_B1);
    }
    if(contrTCC == 300)
    {
        contrTCC=0;
    }
}

void comm_in()
{
    if (i>31)
    {
        return;
    }
    cad_rx[i] = c_rx;
    i++;
}
```

```

void borra_cadena()
{
    for (i=0; i<32;i++)
    {
        cad_rx[i] = 0x00;
    }
    i=0;
}

void leer_prot()
{
    if((cad_rx[0] == 0x7E) &&(cad_rx[2] == ID_NODO))//Busca cabecera
    {
        if(cad_rx[3] == 0x41 && cad_rx[4] == 0x42 && cad_rx[5]==0x52)
        {
            if(cad_rx[8] == 0x41 && cad_rx[9]==0x47 && cad_rx[10]==0x55)
            {
                opt = 0x10;
                borra_cadena();
            }
            borra_cadena();
        }

        if (cad_rx[3] == 0x43 && cad_rx[4] == 0x45 && cad_rx[5] == 0x52)
        {
            if(cad_rx[9] == 0x41 && cad_rx[10] == 0x47 && cad_rx[11] == 0x55)
            {
                opt = 0x20;
                borra_cadena();
            }
            borra_cadena();
        }
        borra_cadena();
    }
    else
    {
        borra_cadena();
    }
}

#int_RTCC
void RTCC_isr(void)
{
    contRTCC++;
    contTAREA++;
}

#int_RDA
void RDA_isr(void)
{
    c_rx = getc();
    comm_in();
}

```

```

.void main()
{

/***** Configuracion del Micro *****/

    setup_adc_ports(NO_ANALOGS);
    setup_adc(ADC_OFF);
    setup_timer_0(RTCC_INTERNAL|RTCC_DIV_32);           //3.26 ms
    setup_wdt(WDT_288MS);
    enable_interrupts(INT_RTCC);
    enable_interrupts(INT_RDA);
    enable_interrupts(GLOBAL);

/***** Inicio del Micro *****/
    rs232_errors = rs232_errors;

    set_tris_c(pto_C);
    set_tris_b(pto_B_sal);
    set_tris_a(pto_A);

    output_low(PIN_C3);
    output_high(PIN_C2);
    output_low(PIN_B0);
    output_low(PIN_B1);
    output_low(PIN_B2);
    output_low(PIN_B3);

    while (TRUE)
    {

        led_tick();

        if (contTAREA > 350)           //Verifica la tarea cada 1ms

        {
            leer_prot();
            contTAREA = 0;
        }

        if (i > 31)
        {
            borra_cadena();
        }

        if (opt == 0x10)
        {
            output_high(PIN_B3);
            output_high(PIN_B0);
            opt = 0x00;
        }

        if (opt == 0x20)
        {
            output_low(PIN_B3);
            output_low(PIN_B0);
            opt = 0x00;
        }

    }
}

```