



Universidad Nacional de Mar del Plata



Facultad de Ingeniería
Departamento de Ingeniería Electrónica

Tesis de Grado
Domótica mediante reconocimiento de voz

Javier Omar Errobidart

Laboratorio de Comunicaciones

Director: Dr. Ing. Alejandro José Uriz
Codirector: Msc. Ing. Esteban Lucio González

Noviembre 2016

Índice de contenidos

Índice de contenidos	1
Resumen.....	3
CAPÍTULO 1	4
1.1 Introducción	4
1.2 ¿Qué es la domótica?.....	4
1.3 Partes de un sistema domótico.....	5
1.3.1 Topología.....	5
1.3.1.1 Arquitectura centralizada.....	6
1.3.1.2 Arquitectura descentralizada	6
1.3.1.3 Arquitectura distribuida	7
1.3.1.4 Arquitectura mixta	7
1.3.2 Protocolo	8
1.3.2.1 X-10	8
1.3.2.2 Konnex.....	11
1.3.2.3 IoT.....	17
1.3.3 Infraestructura de comunicación	19
1.3.3.1 Bluetooth.....	19
1.3.3.2 ZigBee	21
1.3.3.3 WiFi.....	23
1.3.4 Interfaz de usuario	25
1.3.4.1 Modulo VR de Elechouse.....	25
1.3.4.2 SimpleVR de Elechome.....	26
1.3.4.3 EasyVR de Veear.....	27
1.3.4.4 Google Speech Cloud API	28
1.3.5 Sensores y actuadores.....	28
1.4 Conformación del sistema.....	29
1.4.1 Elección de Interfaz de Usuario.....	29
1.4.2 Elección de la Infraestructura de Comunicación.....	30
1.4.3 Elección del Protocolo.....	31
1.4.4 Elección de Topología.....	32
1.4.5 Elección de Periféricos	32
1.5 Conclusiones.....	33
CAPÍTULO 2	35
2.1 Introducción	35
2.2 Características del módulo ESP8266	35
2.2.1 Protocolos WiFi	35
2.2.2 Especificaciones Técnicas.....	36
2.2.3 Características Eléctricas	37
2.2.4 Consumo Energético	38
2.2.5 Características de Radio WiFi	38
2.3 Herramientas de desarrollo	39
2.3.1 Presentaciones comerciales	39
2.3.2 Placa de desarrollo	40
2.3.3 Firmware	41
2.4 Conexión a infraestructura WiFi	42

2.4.1 Rutina de funcionamiento	43
2.4.1.1 Rutina de conexión: Intermediario	44
2.4.1.2 Rutina de conexión: Consultante	46
2.4.1.3 Rutina de difusión	48
2.5 Direccionamiento y sistema de comandos	49
2.5 Conclusiones.....	50
CAPÍTULO 3	52
3.1 Introducción	52
3.2 Interfaz de usuario	52
3.2.1 Gestión de dispositivos	53
3.2.1.1 Flujo de información	54
3.2.1.2 Memoria	57
3.2.2 Menú dinámico	61
3.2.3 Módulo reconocedor de voz	63
3.2.5 Hardware de interfaz ESP8266/Arduino	67
3.3 Dispositivos periféricos	69
3.3.1 Componentes del Hardware	69
3.3.2 Rutina de funcionamiento.....	70
3.4 Conclusiones.....	72
CAPÍTULO 4	73
4.1 Discusión de los resultados	73
4.2 Conclusiones finales	75
4.3 Mejoras del proyecto	75
Bibliografía	77

Resumen

En este informe se presenta el desarrollo del trabajo en calidad de tesis de grado para la Carrera de Ingeniería Electrónica. El proyecto aborda la realización de un sistema de domótica con el cual el usuario podrá operar y llevar un control de los diferentes sistemas que componen un hogar, ya sea mecánicos, eléctricos y electrónicos. Por ejemplo, podrá operar el encendido y apagado de luces, electrodomésticos, levantamiento de persianas, sistemas de riego, entre otros y podrá monitorear variables como temperatura, iluminación, presencia, etc. El sistema reduce el esfuerzo físico que realiza el usuario para llevar a cabo estas tareas, contribuyendo al confort y la autonomía en el caso de personas que sufren algún tipo de invalidez.

El principal objetivo de este proyecto es desarrollar un sistema domótico cuya interfaz sea accesible para usuarios con discapacidades motrices. Para ello se implementará reconocimiento de voz (*Voice Recognition*, VR) como control de un sistema de domótica. Esta iniciativa surge a partir de una experiencia de trabajo con una persona que padece de cuadriplejía, y de conocer sus necesidades.

Las principales ventajas de este proyecto es que provee una mayor versatilidad en cuanto a la interacción con el sistema. Esto es porque que la operación es de manos libres y las opciones del menú se adaptaran dinámicamente al estado del mismo. De esta forma se facilita la integración de nuevos elementos de control. También cuenta con las mismas ventajas tanto para usuarios con diferentes discapacidades motrices como para otro tipo de usuarios. La naturaleza intuitiva de la operación permite acceder a esta tecnología a usuarios que no estén familiarizados con el uso de sistemas informáticos, como pueden ser los adultos mayores.

El equipo que se desarrolló cuenta con tres dispositivos, uno es la interfaz de usuario con el sistema de reconocimiento de voz y los otros dos son periféricos actuadores de dos estados (encendido y apagado), aunque también se admiten sensores y actuadores de múltiples estados. Los equipos se conectan mediante WiFi y los protocolos TCP/IP, para ello requieren de una red anfitriona, es decir que es necesario contar con un punto de acceso WiFi para que los equipos se conecten entre ellos, pero no es necesario que la red tenga acceso a internet.

La interfaz de usuario posee un menú que se muestra en una pantalla LCD, en el cual puede ser navegado mediante comandos de voz. La interfaz de usuario posee también un centro de procesamiento donde se registran los periféricos disponibles para operar y una interfaz WiFi para conectarse a la red. El sistema reconocedor de voz está compuesto por un modulo integrado dedicado para dicho fin, EasyVR de Veeear. La empresa provee importantes herramientas de desarrollo para la plataforma Arduino, es por esa razón se decidió utilizar esta plataforma para la implementación de la interfaz de usuario.

Los dispositivos periféricos están formados por sistemas embebidos que poseen entradas y salidas para manejar tanto actuadores como sensores y también poseen la circuitería y lógica de comunicación para conectarse a WiFi y protocolos de niveles superiores como TCP/IP.

Luego de evaluar las diferentes opciones en cuanto al protocolo de comunicación, se opto por WiFi ya que los microcontroladores que vienen con sistemas embebidos para dicho fin poseen un alto grado de desarrollo y un costo muy competitivo en el mercado. A su vez, esta elección conduce a la utilización de los protocolos TCP/IP en capas superiores lo cual deja el proyecto preparado para una conexión remota vía internet. A pesar de que este proyecto apunta a una completa funcionalidad independiente de internet, se considera que contar con esta posibilidad de conexión es de vital importancia para introducir el proyecto como un producto en el mercado.

CAPÍTULO 1

1.1 Introducción

En este proyecto se propone la realización de un sistema domótico que mejore la calidad de vida de los usuarios con discapacidades motrices. Para lograr este objetivo se plantea una lista de características que se consideran primordiales:

- Debe contar con una interfaz de usuario fácil de utilizar que sea de manos libres.
- Debe ser fácil de integrar al hogar.
- Debe estar preparado para ampliaciones sin grandes modificaciones.

Estos criterios apuntan a una reducción del esfuerzo físico y de la dependencia de terceros para poner en marcha el sistema y su posterior utilización. Estas cuestiones resultan críticas en personas que padecen algún tipo de invalidez.

A su vez se toman como objetivos secundarios:

- Costo final accesible.
- La adecuación del producto en el mercado de consumo masivo.

Un costo final accesible implica que mayor cantidad de usuarios puedan usar el producto. A su vez, para que sea atractivo en el mercado, debe ser fácil de utilizar. En el desarrollo del proyecto se tuvo en cuenta este aspecto a la hora de diseñar los mecanismos de utilización y también se dejará preparado el camino para ampliar fácilmente las funcionalidades del sistema.

El sistema contará con una interfaz de usuario compuesta de un menú navegable a través de comandos de voz. El menú será visualizado en una pantalla de cristal líquido (LCD) el cual contendrá información dinámica de los periféricos disponibles para operar. Los dispositivos periféricos pueden ser sensores u actuadores que son los que finalmente realizarán las tareas que el usuario desea llevar a cabo.

Este Capítulo comienza con una breve introducción acerca de qué es la domótica y que partes constituyen al sistema. Luego se enumeran diferentes alternativas para llevar a cabo cada parte del sistema. Por último se discutirán las opciones y se establecerá un criterio para elegir las partes que finalmente se utilizarán para llevar a cabo el proyecto, ya sea desde un punto de vista tecnológico como conceptual.

1.2 ¿Qué es la domótica?

Se define por domótica al conjunto de sistemas y tecnologías integradas que controlan y automatizan las diferentes instalaciones de un inmueble, contribuyendo a la gestión energética, confort, seguridad, comunicación y accesibilidad entre el usuario y el sistema. [1]

El significado etimológico de domótica sería “casa automatizada”. Edificios y ciudades inteligentes son conceptos similares, pero la palabra domótica se refiere a los hogares de viviendas familiares con sistemas de automatización.

Entre las aplicaciones que brinda la domótica se encuentran:

- **Gestión energética**

Esta abarca el ahorro energético, la eficiencia energética y la generación de energía. Es una de las aplicaciones más importantes de la domótica, ya que sin la inteligencia que aporta no sería posible este tipo de gestión.

- **Confort**

Al poseer el control de todo el inmueble, el mismo se puede centralizar, distribuir, automatizar para adecuarse a las necesidades de cada usuario. Esta versatilidad en el control implica una mejora de la calidad de vida.

- **Seguridad**

Un sistema domótico provee la infraestructura necesaria para proteger las personas y los bienes mediante la prevención y la detección.

La información provista por los sensores es utilizada para realizar acciones programadas para lograr dicho fin.

- **Comunicación**

Es una característica fundamental de la domótica ya que sin ella sería imposible conocer el estado de las variables monitoreadas por los sensores ni tampoco una gestión integrada de parte del usuario. Se puede esquematizar en un extremo la interfaz del usuario y en otro extremo los actuadores y sensores.

La comunicación no se refiere estrictamente a una telegestión o acceso remoto desde fuera del hogar, sino que también abarca las transmisiones internas dentro del inmueble.

- **Accesibilidad**

El sistema domótico permite el acceso de cualquier persona a cualquier entorno. Es la principal característica que persigue este proyecto de trabajo. La accesibilidad busca la facilidad de la gestión del sistema para cualquier tipo de usuario.

1.3 Partes de un sistema domótico

Un sistema domótico puede separarse en cuatro partes fundamentales que son su topología o estructura, la interfaz de usuario, el sistema de comunicación entre las partes y los sensores y actuadores. Cada una de estas partes será analizada, se evaluarán las alternativas de implementación y finalmente se elegirá la más adecuada para la realización del proyecto.

1.3.1 Topología

Es la estructura del sistema. Se refiere a la relación entre la interfaz de usuario, el sistema de comunicación los sensores y actuadores. Establece las formas y roles que tiene cada parte en el sistema y su interacción con las demás.

A grandes rasgos se pueden definir dos tipos de topología, la lógica y la física. La topología lógica es principalmente definida por la arquitectura de control del sistema. Esta depende de si hay uno o varios módulos que son capaces de procesar los datos y de controlar el flujo de la comunicación y de cómo se encaminan los datos.

La topología física tiene que ver con el medio de comunicación utilizado, que puede ser diferentes tipos de cableado o inalámbrico, entre otros. Esto determina qué módulos tendrán comunicación directa entre sí y cuáles no. Por lo tanto este factor resulta condicionante para la inteligencia de comunicación.

A continuación se describen las arquitecturas más utilizadas que dan funcionamiento a diferentes topologías lógicas.

1.3.1.1 Arquitectura centralizada

En este tipo de arquitectura existe un solo controlador el cual coordina la comunicación entre los diferentes elementos del sistema. Esta unidad se considera central ya que en ella reside la inteligencia que permite concretar las acciones solicitadas por el usuario. Sus principales tareas son analizar los datos que envían los dispositivos, procesar las solicitudes y establecer prioridades en la comunicación.

Los elementos periféricos serían la interfaz de usuario, los sensores y actuadores. Estos nunca se comunican entre sí, sino que reciben y transmiten datos desde y hacia la unidad central. Todo el flujo de comunicación pasa por el controlador, por lo tanto, si eventualmente fallara o dejara de funcionar, el sistema en su totalidad quedaría inoperante. Por otro lado, el control de la comunicación es más sencillo en este tipo de arquitectura.

En la Figura 1.1 se muestra un esquema de esta topología.

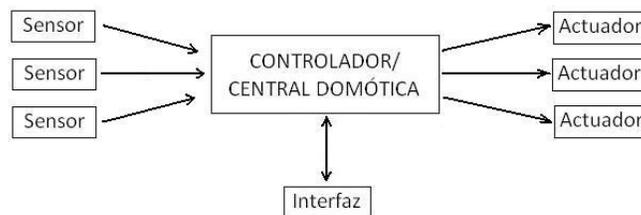


Figura 1.1 Esquema centralizado

1.3.1.2 Arquitectura descentralizada

Esta arquitectura surge como una mejora en términos de robustez de la arquitectura centralizada. En esta los elementos periféricos no tienen el control de la comunicación, al igual que en la arquitectura centralizada, pero pueden coexistir varias unidades centrales o controladores. De esta forma, si falla alguna unidad central el sistema sigue operando mientras haya al menos un controlador funcionando. En la Figura 1.2 se plantea un ejemplo de este tipo de arquitectura.

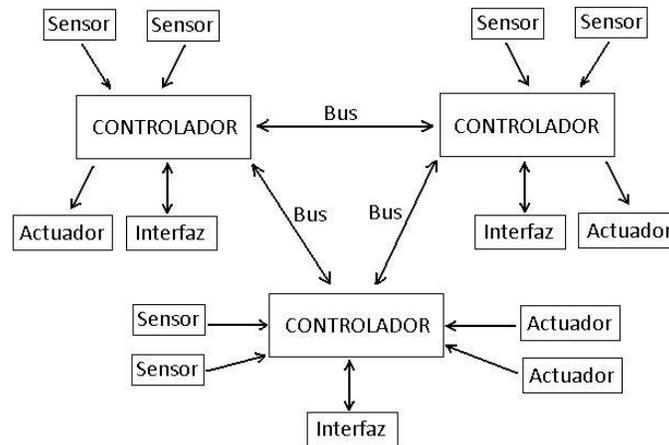


Figura 1.2 Esquema descentralizado

1.3.1.3 Arquitectura distribuida

En esta arquitectura cada sensor, actuador e interfaz de usuario es también un controlador de la comunicación. No existe una unidad central, la inteligencia de control está distribuida y el rol de control va variando de dispositivo.

Es mucho más robusto que un sistema con arquitectura centralizada o descentralizada, pero la desventaja es que el costo es más elevado debido a la complejidad. El esquema de esta arquitectura se presenta en la Figura 1.3.

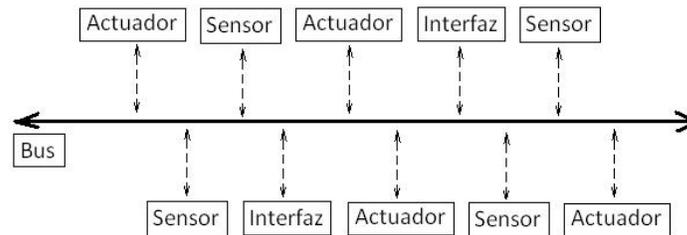


Figura 1.3 Esquema distribuido

1.3.1.4 Arquitectura mixta

En este caso se combinan las otras arquitecturas. Puede haber uno o varios controladores dedicados. También algunos sensores y actuadores pueden llegar a tomar el control de la comunicación, mientras otros pueden estar exentos de esta capacidad.

Normalmente se estructura por zonas. Por ejemplo, puede haber subredes centralizadas o distribuidas que se conectan a su vez a otra red de mayor jerarquía. La Figura 1.4 presenta un esquema de esta topología.

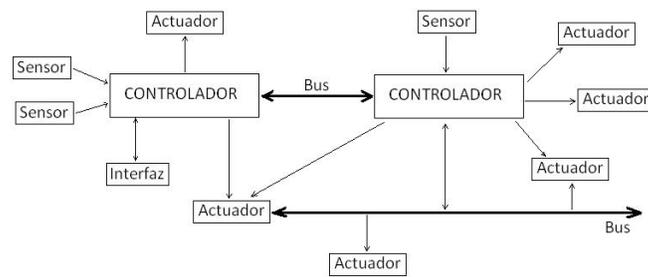


Figura 1.4 Esquema mixto

La arquitectura utilizada debe tener el soporte de una inteligencia de comunicación compatible. Para ello existen diversos protocolos de comunicación que se pueden aplicar al sistema. También debe contar con una infraestructura física que permita el funcionamiento, lo que previamente se ha denominado topología física.

1.3.2 Protocolo

Un protocolo describe un método por el cual se lleva a cabo la comunicación entre las diferentes partes en un sistema, en este caso domótico. Además de definir la lógica por la cual se realiza la comunicación, también define métodos de corrección de errores y establece requisitos en la infraestructura del hardware para garantizar el funcionamiento correcto del sistema.

1.3.2.1 X-10

Este protocolo fue desarrollado por PicoElectronics, en Escocia entre 1976 y 1978. La idea que impulsó su desarrollo fue permitir el control remoto de dispositivos domésticos, de esta forma este protocolo se convirtió en pionero de la domótica. Su principal característica es que utiliza el cableado eléctrico como bus de comunicación. [2]

El producto fue apuntado al segmento hogares, para ello se comercializaron módulos autoinstalables, que se colocan directamente en las bocas de distribución eléctrica del hogar (220v/110v y 50Hz/60Hz). Esta facilidad permitió una buena penetración en el mercado, logrando una madurez de la tecnología durante el transcurso de los años. Hoy en día, X-10 ha logrado alcanzar precios muy competitivos, permitiéndole ser líder en el mercado norteamericano residencial y de pequeñas compañías.

El protocolo permite la utilización de más de un controlador que puedan operar en simultaneo los distintos actuadores, por lo tanto tiene una arquitectura descentralizada, como se describió anteriormente. Los dispositivos se agrupan mediante un "Código de Casa" que es una letra de la A hasta la P, que representaría la habitación o una función que tienen en común. En cada grupo se pueden direccionar 16 dispositivos mediante el "Numero de Módulo" que va del 1 al 16. Por lo tanto, se pueden direccionar un total de 256 módulos en toda la casa. Cada dispositivo posee dos ruedas seleccionadoras, una roja para seleccionar el código de casa y una negra para elegir el numero de modulo. Este sistema facilita la instalación y configuración. El direccionamiento queda a criterio del usuario el cual puede adaptarlo a sus necesidades y cambiarlo cuando lo necesite. La Figura 1.5 muestra un dispositivo de este tipo. [3]

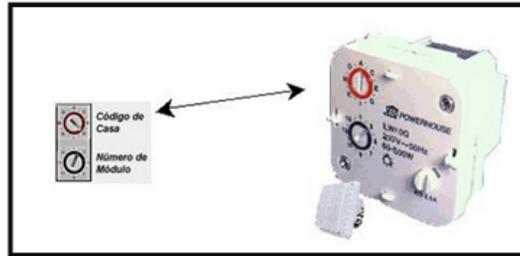


Figura 1.5 Modulo X-10

Las señales enviadas son recibidas por todos los módulos, pero solo actuará el que tiene la misma dirección que posee el mensaje. Es posible asignar una misma dirección a varios módulos, de forma que cuando se envíe un comando a esa dirección actuaran todos esos dispositivos. Un esquema de una red X-10 es mostrado en la Figura 1.6.

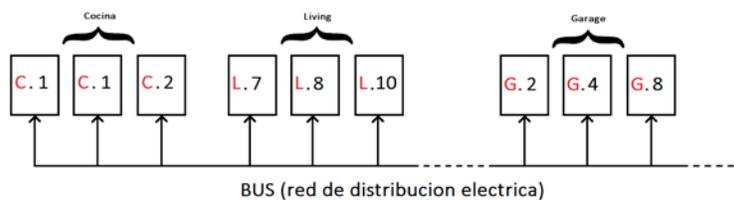


Figura 1.6 Ejemplo de red X-10

Las señales de control son transmitidas en ráfagas de RF a 120KHz que duran 1ms, sincronizadas con los cruces por cero de la tensión alterna de línea. La presencia de una ráfaga en un cruce por cero representa un 1 lógico, y la ausencia representa el 0 lógico. Cada bit transmitido se debe enviar con su complemento en el cruce por cero del siguiente semiciclo. La razón de esta redundancia es con el fin de reducir errores. A su vez, cada mensaje es transmitido dos veces. Por lo tanto el protocolo cuenta con cuádruple redundancia. La Figura 1.7 muestra la señal X-10 solapada a la de red eléctrica.

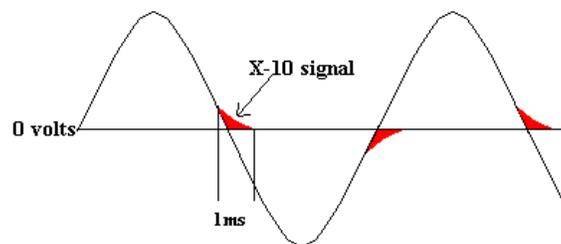


Figura 1.7 Señal X-10

Como en cada ciclo hay dos cruces por cero se envían dos bits por ciclo. Pero debido a la redundancia el sistema toma como información útil solo un bit por ciclo. Entonces la velocidad de transmisión queda impuesta por la frecuencia de línea. Si es de 50Hz tendremos una tasa de 50bps y si es de 60Hz será de 60bps.

Al principio de cada mensaje se envía un código de inicio para indicar el comienzo de la trama. Este código es 1110 y se envía sin la redundancia por bit, así que el envío se lleva a cabo en dos ciclos. Luego se envía el código de casa compuesto de 4 bits con su respectiva redundancia, esto se lleva a cabo en 4 ciclos. Finalmente, dependiendo del tipo de mensaje, se enviará el número de módulo o el código de función. Ejemplos de comandos X-10 son mostrados en la Figura 8.

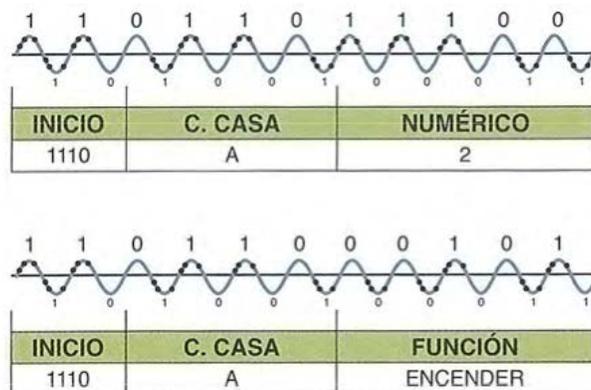


Figura 1.8 Transmisión X-10

En un comienzo, los únicos comandos que admitía el protocolo eran “encender”, “apagar”, “aumentar”, “disminuir”, “todo apagado” y “todo encendido”. El tamaño de estos mensajes es de 48 bits. Posteriormente se expandieron los comandos a 256 diferentes, mediante el uso de una cabecera especial se pueden superar los 48 bits de tamaño del mensaje.

Los módulos originales eran unidireccionales, algunos enviaban información y otros solo la recibían, luego se implementaron dispositivos bidireccionales que permitían responder los mensajes confirmando la correcta realización de las acciones. Esto permitió implementar aplicaciones en computadora para monitorizar el estado de los diferentes dispositivos en el hogar. Los comandos X-10 son los siguientes.

Código	Función	Descripción	Unidireccional	Bidireccional
0 0 0 0	All units off	Apaga todos los dispositivos con el código de casa indicado en el mensaje	X	
0 0 0 1	All lights on	Enciende todas las luces (con la posibilidad de controlar el brillo)	X	
0 1 1 0	All lights off	Apaga todas las luces	X	
0 0 1 0	On	Enciende un aparato	X	
0 0 1 1	Off	Apaga un aparato	X	
0 1 0 0	Dim	Atenúa la intensidad de la luz	X	
0 1 0 1	Bright	Incrementa la intensidad de la luz	X	
0 1 1 1	Extended code	Código de extensión		X
1 0 0 0	Hail request (solicitud de saludo)	Solicita una respuesta del dispositivo(s) con el código de casa indicado en el mensaje		X
1 0 0 1	Hail acknowledge (confirmación de saludo)	Respuesta al comando anterior		X
1 0 1 0	Pre-set dim	Permite la selección de dos niveles predefinidos de intensidad de luz		X
1 1 0 1	Status is on	Respuesta a la Solicitud de Estado indicando que el dispositivo está encendido		X
1 1 1 0	Status is off	Respuesta indicando que el dispositivo está apagado		X
1 1 1 1	Status request	Solicitud pidiendo el estado de un dispositivo		X

Tabla 1.1 Comandos X-10

Las principales ventajas del sistema X-10 es la fácil integración a un hogar ya edificado y en funcionamiento, debido a que no necesita la instalación de ningún BUS adicional por compartir el mismo cableado que la distribución eléctrica. Por su sencillez, permite que el mismo usuario lleve a cabo la instalación, mantenimiento y ampliación del sistema. Además los módulos tienen un precio muy competitivo, pudiendo automatizar una vivienda a muy bajo costo.

La desventaja de este sistema es que no se puede utilizar en edificios industriales o en edificios con fuertes perturbaciones electromagnéticas, puesto a que las cargas inductivas y capacitivas producen ruido en la red eléctrica y perjudica la transmisión de datos, deviniendo un funcionamiento defectuoso.

1.3.2.2 Konnex

Es un estándar de protocolo de comunicación de red basado en el modelo OSI¹ para edificios inteligentes, tanto para hogares como para otros inmuebles de mayor tamaño (edificios públicos, empresas, instituciones, etc.). El protocolo está basado en tres estándares previos, EHS, EIB y BatiBUS². Esta iniciativa es de origen europeo y amparado por la Unión Europea con el fin de disminuir las importaciones de productos del mismo tipo provenientes de los mercados japoneses y norteamericanos.

La especificación está basada en la pila de comunicación de EIB completada con los mecanismos de configuración, medios físicos y experiencia de aplicación originalmente desarrollados por BatiBUS y EHS. [4]

En el sistema KNX la transmisión de datos se realiza por un medio al cual están conectados todos los dispositivos, permitiendo la intercomunicación entre estos. De esta forma es posible que cualquier componente de órdenes a cualquier otro independientemente de la distancia entre ellos y la ubicación. [6]

Los medios físicos que utiliza el protocolo Konnex son:

- Par trenzado (TP1) aprovechando la norma EIB equivalente.
- Par trenzado (TP0) aprovechando la norma BatiBUS equivalente.
- Portadora sobre línea eléctrica (PL100) aprovechando la norma EIB equivalente.
- Portadora sobre línea eléctrica (PL132) aprovechando la norma EHS equivalente.
- Ethernet aprovechando la norma EIB.net.
- Radiofrecuencia aprovechando la norma EIB.RF.

Cada medio tiene un alcance y una velocidad de transmisión diferente, y a su vez, puede que no sea compatible con otros medios de comunicación presentes en el edificio. Estas características deben tenerse en cuenta al momento de elegir el medio de comunicación, de forma que el protocolo pueda adaptarse a las instalaciones y a las necesidades.

Para interconectar los dispositivos se permite cualquier topología ya sea árbol, estrella, bus y anillo. Solamente no se permitirá cerrar anillos entre líneas situadas topológicamente en diferentes áreas.

El protocolo establece una red jerarquizada, donde la unidad mínima es una línea. En esta se puede conectar hasta un máximo de 64 dispositivos. Si se unen varias líneas a una línea maestra se construye lo que se denomina un "área". De esta línea principal o maestra pueden salir hasta un máximo de 15 líneas secundarias o esclavas. Esto implica un total de 960 dispositivos por área.

Cada línea secundaria debe contar con un acoplador para insertarse en la línea principal. También es posible que las líneas secundarias tengan diferentes medios de transmisión entre si y entre la línea

1 El modelo OSI es un modelo de referencia para protocolos de red con arquitectura en capas. [5]

2 Son protocolos de domótica que brindan especificaciones para diferentes capas del modelo OSI.

principal, por lo tanto el acoplador hace las veces de interfaz entre diferentes medios. Un esquema de una red Konnex es mostrado en la Figura 1.9.

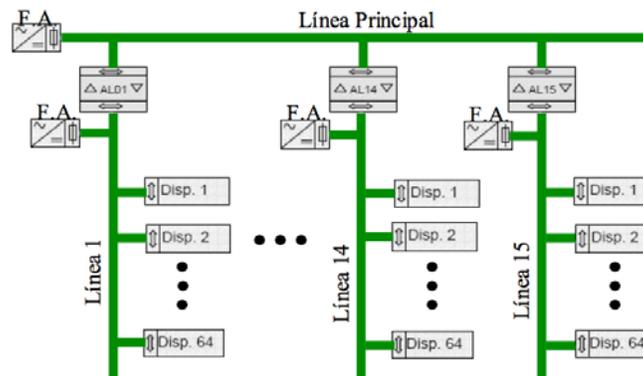


Figura 1.9 Interconexión de líneas Konnex

El *backbone* es la línea principal de mayor jerarquía, a la cual se pueden conectar hasta 15 áreas. Esto da un total de 14400 dispositivos en todo el sistema. Las líneas maestras de cada área se conectan mediante un acoplador a la línea de *backbone*. De no ser suficiente esta cantidad de dispositivos se puede ampliar la capacidad de cada línea mediante diferentes técnicas, por ejemplo, el uso de repetidores. La Figura 10 presenta un ejemplo de un *Backbone*.

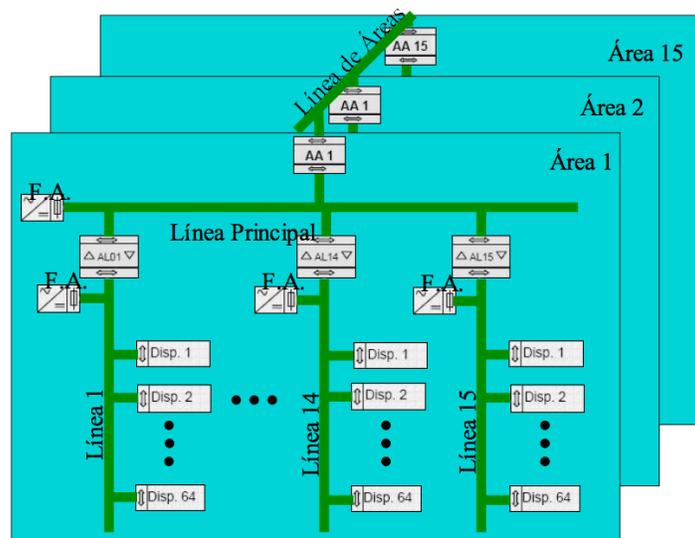


Figura 1.10 Interconexión de áreas Konnex

Cada dispositivo cuenta con una dirección física de 16 bits que le permite ser identificado unívocamente en todo el bus. La dirección se divide en tres partes, los primeros 4 bits identifican el área o zona que, como ya se mencionó antes, son 15. Los siguientes 4 bits identifican la línea. Y los siguientes 8 bits determinan la dirección del componente. La dirección 0 se reserva para direccionar la propia línea con intenciones de *multicast*.

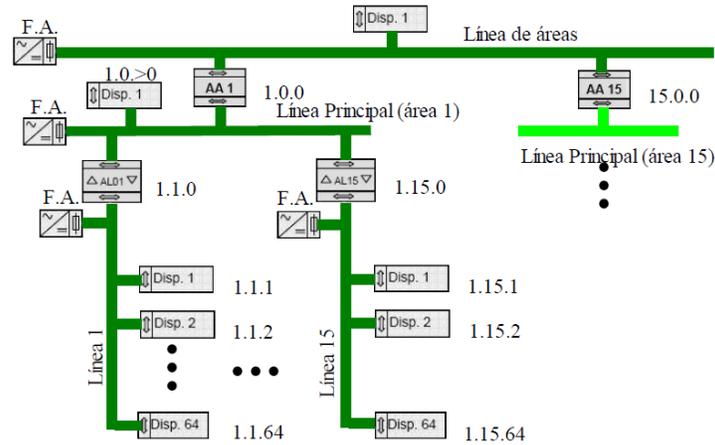


Figura 1.11 Esquema de direccionamiento Konnex

Existe otro tipo de dirección asociada a cada dispositivo que es independiente de la topología del bus y que corresponde a una capa más elevada en el modelo OSI. La dirección de grupo se encarga de definir funciones específicas del sistema, estableciendo relaciones entre dispositivos y agrupándolos. Por ejemplo, un sensor de luminosidad puede enviar una señal a una dirección de grupo, de esta forma está indicando qué dispositivos actuarán. En este grupo pueden incluirse todos los actuadores que afectan la variable luminosidad, que pueden ser luces, persianas, etc. De acuerdo a la aplicación, el sensor puede hacer *multicast* directo sobre los dispositivos que regulan la variable en cuestión.

Esta división en grupos permite asociar funcionalmente los dispositivos. Es ahí donde radica la inteligencia que permite a este protocolo actuar de forma distribuida. Las direcciones de grupo relacionan los elementos de entrada, que son los sensores, con los elementos de salida, que son los actuadores. Las instrucciones son enviadas selectivamente desde los sensores hacia los dispositivos que de un grupo determinado.

La dirección de grupo está formada por 15 bits. Se puede emplear de dos formas: direccionamiento a dos niveles y a tres niveles.

En el direccionamiento a dos niveles los primeros 4 bits representan el grupo principal y los 11 bits restantes el subgrupo. En el direccionamiento a tres niveles los primeros 4 bits es el grupo principal, los siguientes 3 bits es el grupo medio y los último 8 bits el subgrupo. Ejemplos de direcciones de grupo son mostrados en la Figura 12.

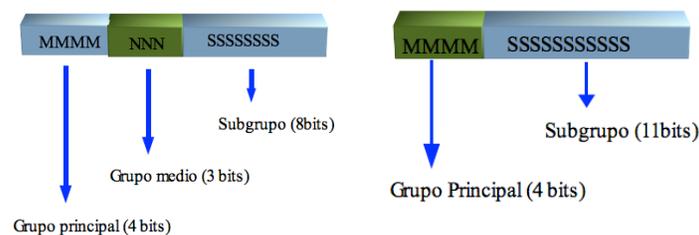


Figura 1.12 Direcciones de grupo

En ambos casos el grupo principal se usa para diferenciar grupos funcionales, por ejemplo, iluminación, seguridad, temperatura, etc. Se usan los valores del 1 al 13, los valores 14 y 15 no deben usarse

ya que son filtrados por los acopladores. La dirección todos ceros (0/0/0) está reservada para uso del sistema.

Como este direccionamiento funciona en una capa superior del modelo OSI respecto a las direcciones físicas, es posible abstraernos de estas. Así se pueden relacionar equipos que están en diferentes líneas o aéreas, solo se debe tener en cuenta las siguientes consideraciones:

- Los sensores pueden tener asociada una sola dirección de grupo.
- Varios actuadores pueden tener asociada la misma dirección de grupo.
- Un mismo actuador puede tener asociada diferentes direcciones de grupo.

De esta forma, los sensores quedarán direccionados en grupos que siempre serán diferentes a los de los actuadores. Estos últimos, sin embargo, pueden agruparse de forma conveniente para la funcionalidad del sistema.

En el sistema Konnex existen varios medios de transmisión, en los cuales el acceso al medio varía en función de estos. Para el caso de EIB en par trenzado se usa CSMA/CA y para EIB en radiofrecuencia CSMA/CD.

Cuando se envía una trama, si el BUS no está ocupado, los dispositivos receptores deben enviar un acuse de recibo. Si la información llega de forma incorrecta se reenvía la trama. Este proceso se puede repetir hasta un máximo de tres veces. Si el BUS está ocupado, el dispositivo que debe transmitir esperara hasta que el BUS se despeje. Para optimizar el BUS, los acopladores pueden bloquear la propagación de tramas hacia líneas o aéreas a los cuales no va dirigido el mensaje. De esta forma se reduce la cantidad de mensajes en circulación.

El formato de las tramas Konnex en la norma para par trenzado es el heredado de la norma EIB. Los datos se transmiten en modo simétrico. Además, el par trenzado provee el balance que garantiza que las interferencias electromagnéticas afectarán a los dos conductores de la misma forma. Luego, cada par de vueltas en la trenza, la interferencia se cancela.

La transmisión es binaria y se realiza en banda base. El "1" lógico se representa mediante la ausencia de impulso, mientras que el "0" lógico se representa mediante un pulso negativo que luego se pasa a positivo, esto es mostrado en la Figura 1.13.

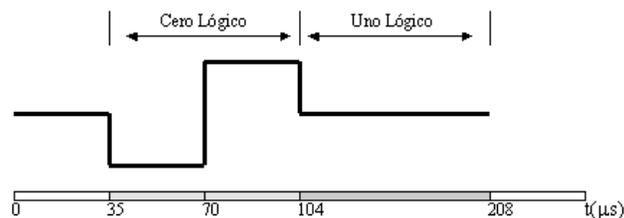


Figura 1.13 Transmisión binaria en banda base

El formato de la trama no varía por el medio físico en el cual se transmite. La transmisión es asíncrona. Cuando un dispositivo se dispone a transmitir, verifica durante un tiempo t_1 que el BUS esté disponible, luego envía el mensaje. Después espera durante un tiempo t_2 el acuse de recepción de la trama enviada. En el caso que no se reciba, o se reciba con errores, el proceso se repetirá hasta un máximo de tres veces.

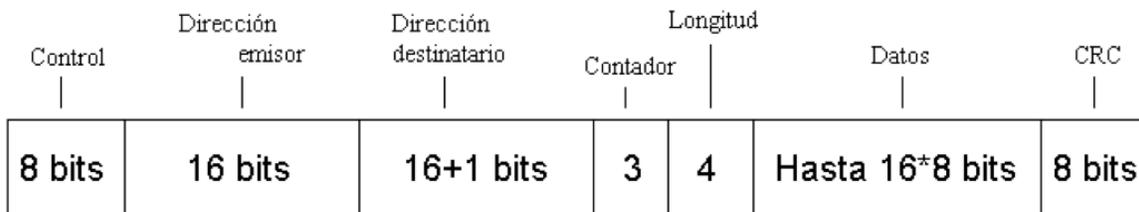


Figura 1.14 Trama de comunicación Konnex

Control: En este campo se especifica la prioridad del mensaje y si es la primera vez que se envía el mensaje o es un reitero.

Origen: Es la dirección física del dispositivo que envía la trama. Por lo tanto posee los bits que detallan el área, la línea y el número de dispositivo.

Destino: Aquí se especifica la dirección del dispositivo de destino, la misma se puede representar de dos formas. Si el bit de mayor peso es un "0", la dirección especificada es la dirección física del dispositivo de destino. Si el bit de mayor peso es un "1" la dirección representada es de grupo y puede ir dirigida a uno o varios dispositivos.

Long: Indica la cantidad de bits del campo de "datos útiles".

Datos Útiles: Estos son los datos que contienen la información que los dispositivos interpretarán para realizar acciones. Los datos siguen el estándar EIS (EIB *Interworking Standart*). Este estándar define 15 tipos de datos diferentes y cada uno está asociado a un tipo de acción de control.

Comprobación: Permite comprobar que la trama recibida es correcta. En el caso de ser correcto se enviara un asentimiento ACK, de lo contrario se enviará un asentimiento negativo NACK. Si el receptor está ocupado, puede enviar una trama colocando en este campo la bandera de BUSY, indicando su estado al dispositivo que quiere transmitirle mensaje para que lo intente luego.

Para que los dispositivos puedan intercambiar información y que esta pueda ser interpretada por todos, sin importar el fabricante, se debe emplear la misma semántica. Para eso el sistema Konnex utiliza el EIB *Interworking Standart* (EIS). El tamaño del mensaje es variable y existen diferentes tipos, *switching*, *dimming*, datos, etc. En la Tabla 1.2 se muestran los comandos del estándar.

Funcion KNX/EIB	Tamaño	EIS	DPT	Value
Switch	1 Bit	EIS 1	DPT 1	0,1
Dimming (Position, Control, Value)	1 Bit, 4 Bit, 8 Bit	EIS 2	DPT 3	[0,0]...[1,7]
Time	3 Byte	EIS 3	DPT 10	
Date	3 Byte	EIS 4	DPT 11	
Floating point	2 Byte	EIS 5	DPT 9	-1341849,6
8-bit unsigned value	1 Byte	EIS 6	DPT 5	0...255
8-bit unsigned value	1 Byte	DPT 5.001	DPT 5.001	0...100
Blinds / Roller shutter	1 Bit	EIS 7	DPT 1	0,1
Priority	2 Bit	EIS 8	DPT 2	[0,0]...[1,1]
IEEE Floating point	4 Byte	EIS 9	DPT 14	4-Octet Float Value IEEE 754
16-bit unsigned value	2 Byte	EIS 10	DPT 7	0...65535
16-bit signed value	2 Byte	DPT 8	DPT 8	-32768...32767
32-bit unsigned value	4 Byte	EIS 11	DPT 12	0...4294967295
32-bit signed value	4 Byte	DPT 13	DPT 13	-2147483648...2147483647
Access control	1 Byte	EIS 12	DPT 15	
ASCII carácter	1 Byte	EIS 13	DPT 4	
8859_1 character	1 Byte	DPT 4.002	DPT 4.002	
8-bit signed value	1 Byte	EIS 14	DPT 6	-128...127
14 character ASCII	14 Byte	EIS 15	DPT 16	
14 character 8859_1	14 Byte	DPT 16.001	DPT 16.001	
Scene	1 Byte	DPT 17	DPT 17	0...63
HVAC	1 Byte	DPT 20	DPT 20	0...255
Unlimited string 8859_1	.	DPT 24	DPT 24	
List 3-byte value	3 Byte	DPT 232	DPT 232	RGB[0,0,0]...[255,255,255]

Tabla 1.2 Comandos de EIB Interworking Standart (EIS).

Además de los elementos pasivos, como fuentes de alimentación y el propio BUS, Konnex es formado por dispositivos que poseen una cierta inteligencia que dota al sistema de sus características. La norma también detalla la arquitectura que poseerá cada uno de estos dispositivos inteligentes.

A continuación se analiza las ventajas y desventajas del sistema Konnex.

Ventajas:

- Posee gran flexibilidad, permitiendo adaptar el sistema a diferentes tipos de edificios, ya sea viviendas familiares o edificios de mayor envergadura. También permite ampliación.
- Se pueden utilizar dispositivos de diferentes fabricantes.
- Proyecto e instalación sencilla.
- Permite alcanzar una mayor tasa de transmisión al tratarse de un BUS específico para transmitir datos.

- Es conveniente para edificios y hogares de nueva construcción, ya que el costo de instalar el cableado adicional para el BUS es menor que si la construcción ya está hecha, y las ventajas que brinda justifican el gasto.
- Es menos sensible a perturbaciones electromagnéticas.
- Puede intercomunicarse con otros sistemas de gestión de edificios.
- Tiene conexión a computadoras para planificación y mantenimiento, como así también a las redes de telecomunicación.

Desventajas:

- Precisa elementos adicionales para comunicarse con el sistema, esto eleva los costos.
- Al poseer una arquitectura de control distribuida, el costo de cada elemento del sistema es más elevado respecto a un sistema centralizado o descentralizado.

1.3.2.3 IoT

Internet de las cosas (en inglés de *Internet of Things*) es un enfoque emergente de internet el cual apunta a la interconexión de objetos cotidianos. [7]

Internet surgió conceptualmente como un medio de comunicación digital entre ordenadores. La principal función de estos dispositivos es la de almacenamiento y gestión de información, además de servir como herramienta de cálculo. Posteriormente, con la integración de contenidos multimedia, se introdujeron aplicaciones de entretenimiento y producción artística entre otros. Por lo tanto, la comunicación de ordenadores implica la comunicación de este tipo de información cuya fuente son los mismos usuarios, las personas.

El enfoque clásico de internet sería el del internet de las personas, ya que en la red circulan ideas, información y servicios que provienen de los usuarios de los ordenadores. Foros, chats, blogs, videoconferencias, tiendas online, bancos, y demás aplicaciones en la web comunican personas con personas que transmiten ideas o brindan servicios. Esta información, en algún momento, tuvo que ser introducida por una persona mediante algún medio de digitalización para que estuviera disponible en la red.

En el internet de las cosas, la información que circula es digitalizada y puesta online por los mismos objetos, sin ser necesaria la intervención de una persona. Estos objetos no serían coordinados por ninguna central u ordenador, sino que serían independientes y tendrían una dirección única en internet para poder ser accedidos. A su vez serían codificados, ya que los parámetros y variables de interés estarían digitalizados y puestos a disposición en la red para algún agente que lo requiera.

Para cumplir con estas funciones, el objeto deberá contar con algún sistema embebido que permita digitalizar la información de interés y subirla a internet. Gracias al avance de la microelectrónica se han desarrollado poderosos microcontroladores que manejan sensores, actuadores y también protocolos de comunicación a diferentes niveles para poder conectarse directamente a internet sin la necesidad de un ordenador.

Es importante destacar que internet de las cosas conduce a una transformación del objeto hacia la red. Al direccionarlo y codificarlo permite que un usuario pueda interactuar con dicho objeto de una nueva forma, y a su vez, aumenta la capacidad de interacción con otros agentes que de otra forma resultaría imposible, como ordenadores y objetos a distancia.

IoT y Domótica son dos conceptos totalmente diferentes pero poseen un punto de convergencia.

La domótica abarca más que la accesibilidad, también apunta a una autogestión por medio de sistemas inteligentes para aumentar el confort y seguridad del hogar. Para cumplir con estos objetivos no necesariamente se debe poseer una accesibilidad remota. Pero en la actualidad, con la importante penetración de los dispositivos móviles con conexión a internet, el acceso remoto por este medio implica un aumento de confort en la vida de las personas. IoT ofrece acceso remoto y deja a disposición del proyectista el objeto codificado en la nube para transformar la interacción con la versatilidad que brindan las diferentes formas de acceso a internet.

Si bien Internet de las Cosas no es un protocolo como se viene describiendo en los apartados anteriores, se presenta como un enfoque y una plataforma sobre la cual llevar a cabo un sistema de domótica.

Ventajas:

- Al utilizar los protocolos TCP/IP y UDP los dispositivos IoT son compatibles con gran cantidad de aplicaciones.
- Promueve la independencia de cada dispositivo, sin necesitar de un coordinador o de otras unidades, solo requiere estar conectado a WiFi.
- Si la red WiFi anfitriona posee conexión a internet, cada dispositivo IoT se conecta a internet sin la necesidad de ningún agente adicional.
- Al conectarse por WiFi posee compatibilidad directa con equipos que estén conectados por Ethernet.
- Poseen un bajo costo.

Desventajas:

- Solo son compatibles con Ethernet. Para otros sistemas de comunicación o protocolos necesitan de una interfaz.
- Poseen un consumo de energía relativamente alto en comparación con otros dispositivos de comunicación inalámbrica.

1.3.3 Infraestructura de comunicación

▪ **Cableado Exclusivo**

En este caso se instala un cableado por todo el hogar con el fin de conectar el sistema domótico. Sin importar si existen otros tendidos de cable en la casa, estos no se utilizan para la comunicación y son independientes del sistema. Es una opción costosa y suele incluirse desde el proyecto de un hogar que aun no se ha construido, es poco práctico para el caso de casas que ya están edificadas y no contemplaron este cableado adicional.

▪ **Cableado Compartido**

Es solución más conveniente en los casos de agregar un sistema domótico a una casa edificada ya que se usa un cableado existente, normalmente el de distribución de energía eléctrica.

Para la transmisión de datos en este medio, se modula una portadora que se encuentra dentro de los 20kHz y los 200kHz. A esta técnica se la denomina *Power Line Carrier* (PLC) y se aplica directamente sobre el cableado eléctrico.

▪ **Inalámbrico**

Una comunicación inalámbrica entre las partes de un sistema domótico provee una mayor versatilidad que un sistema cableado en términos de movilidad y a la hora de ampliar el alcance del mismo. Existen varios protocolos y tecnologías inalámbricas que pueden utilizarse. En las siguientes secciones se estudiarán sus características.

1.3.3.1 Bluetooth

Es una especificación industrial para redes inalámbricas de área personal (WPAN) que posibilita la transmisión de voz y datos. El objetivo de esta norma es eliminar el uso de cables, vincular equipos móviles, crear pequeñas redes y facilitar la sincronización de datos entre dispositivos personales. [8]

Este protocolo de comunicaciones está orientado hacia dispositivos móviles, por lo tanto considera un bajo consumo de energía para el funcionamiento. Como se pretende sustituir cables de datos por una conexión inalámbrica, basta con un corto alcance de emisión. Esto se puede implementar mediante el uso de transceptores de bajo costo.

Los dispositivos se clasifican según su potencia de transmisión, el alcance de cada clase puede variar dependiendo de la existencia de obstáculos entre emisor y receptor. En la Tabla 1.3 se pueden ver las clases y las potencias que transmite cada una.

Clase	Potencia máxima permitida (mW)	Potencia máxima permitida (dBm)	Alcance (aproximado)
Clase 1	100 mW	20 dBm	~100 metros
Clase 2	2.5 mW	4 dBm	~5-10 metros
Clase 3	1 mW	0 dBm	~1 metro

Tabla 1.3 Clases de dispositivos Bluetooth

Todas las clases son compatibles entre sí. Cuando un dispositivo es de menor clase no solo aumenta la potencia de emisión, sino también la sensibilidad. Eso le permite detectar dispositivos de clases más altas (menos alcance) y adaptar la potencia de emisión para ser detectable por aquel dispositivo.

La tasa de transmisión de datos, o ancho de banda, fue mejorada con el transcurso del tiempo debido a subsecuentes actualizaciones del protocolo. A cada una de ellas se le asignó un número de versión. Esto derivó a una nueva clasificación de dispositivo según su ancho de banda de transmisión. En la Tabla 1.4 se muestran las diferentes versiones con la tasa de transferencia máxima que puede alcanzar.

Versión	Ancho de banda
Versión 1.2	1 Mbit/s
Versión 2.0 + EDR	3 Mbit/s
Versión 3.0 + HS	24 Mbit/s
Versión 4.0	32 Mbit/s

Tabla 1.4 Versiones de Bluetooth

La transmisión de Bluetooth opera en la banda de 2,4GHz a 2,48GHz (banda ISM), utiliza modulación de amplio espectro con salto de frecuencia con posibilidad de transmitir en full dúplex con máximo de 1600 saltos por segundo.

El protocolo de banda base combina conmutación de circuitos y paquetes. La conmutación de circuitos puede ser asíncrona o síncrona. Cada canal permite soportar tres canales de datos síncronos (voz) o un canal de datos síncrono y otro asíncrono.

Uno de los principales usos de Bluetooth es la simplificación del descubrimiento y configuración de dispositivos. Este servicio puede ser utilizado por diversas aplicaciones para establecer rápidamente una conexión.

El protocolo permite establecer punto a punto o punto a multipunto. Las pequeñas redes que se forman son llamadas "*piconet*" y están estructuradas mediante el esquema maestro y esclavo. Solo un dispositivo es maestro el cual puede conectarse a uno o varios dispositivos esclavos. En una *piconet*, todos los dispositivos están conectados en el mismo canal. A medida que se van sumando dispositivos esclavos los recursos de la red se comparten y la tasa de transferencia neta en cada conexión entre dos dispositivos decrece. Como solución a este problema se habilitó el uso de las llamadas *scatternet*, que son la unión de dos hasta diez *piconets*. Así, las diferentes *piconets* se establecen en distintos canales. Si bien el rendimiento sigue decreciendo a mayor cantidad de dispositivos, en conjunto e individualmente se obtiene mejores tasas de transferencia que si todos los dispositivos estarían conectados a una misma *piconet*.

Ventajas:

- El método para enlazar dispositivos es muy sencillo.
- Debido a la penetración en el mercado y su amplia utilización posee un bajo costo.
- Funcionan con un bajo consumo de energía.
- Permite tasas de transferencia aceptables.
- Puede coexistir con redes WiFi debido al salto en frecuencia que utilizan ambas tecnologías.

Desventajas:

- La cantidad de dispositivos que se pueden conectar se limita a 8.
- El alcance máximo que ofrece es de 10 metros.
- Utiliza la banda ISM a 2,4GHz la cual se encuentra muy saturada y la presencia de otros equipos que trabajen en esa frecuencia puede alterar el funcionamiento del enlace.
- La tasa de transferencia y el rendimiento general de la red disminuye a mayor cantidad de dispositivos conectados.

1.3.3.2 ZigBee

Es la especificación de un conjunto de protocolos de comunicación inalámbrica de alto nivel para transmisores digitales de radiofrecuencias. Este protocolo se basa en el estándar IEEE 802.15.4 de redes WPAN (*Wireless personal area network*). El principal objetivo son las aplicaciones para redes *Wireless* que requieran comunicaciones seguras con baja tasa de envío de datos y mucha más vida útil de las baterías. [9]

Propiedades	Rango
Transmisión de datos	868 MHz:20kb/s ; 915MHz: 40kb/s ; 2.4GHz: 250kb/s
Alcance	10-20m
Latencia	Por debajo de 15ms
Canales	ISM 868/915MHz; 11 canales. 2.4GHz: 16 canales
Banda de frecuencia	Dos PHY: 868/915MHz y 2.4GHz
Direccionamiento	Cortos de 8 bits o 64 bits IEEE
Canal de acceso	CSMA-CA y CSMA-CA ranurado
Temperatura	-40 a +85 grados centígrados

Tabla 1.5 Detalles técnicos de ZigBee

Los dispositivos que intervienen en la formación de una red ZigBee pueden formar redes de topología estrella, árbol y malla. Esta última resulta ser la que aprovecha mejor las características del estándar.

En la topología en malla, si un camino eventualmente deja de estar disponible, se forman nuevos a través de los otros dispositivos. Estos actúan como nodos y cada uno puede realizar un rol diferente en la comunicación.

Además, los dispositivos están dotados de determinadas características para cumplir con la aplicación del usuario. Estas diferencias dan lugar a dos tipos de clasificación para los dispositivos.

➤ Según su función en la comunicación de la red:

- **Coordinador:** Debe existir al menos uno por red. Sus funciones son las de encargarse de controlar la red y los caminos que deben seguir los dispositivos para conectarse entre ellos. Si un dispositivo cae, el coordinador es el responsable de gestionar los nuevos caminos. Para poder cumplir con estas tareas reúne un perfil de hardware más exigente que el resto de los dispositivos.
- **Enrutador:** Es un dispositivo intermediario en la comunicación entre otros. De esta forma establece un camino para la transmisión de los datos. El mensaje puede pasar por varios enrutadores antes de llegar a destino. El que establece por cuales de estos pasara el mensaje es el dispositivo coordinador.

- Dispositivo Final: Este solo se comunica con un “nodo padre” que puede ser el coordinador o un enrutador y no se comunica con ningún otro dispositivo. Esto le permite estar en modo bajo consumo por mayor tiempo y aumentar significativamente la vida útil de la batería. Debido a estas características, el hardware que requiere es menos exigente y resulta ser más económico que los otros tipos de dispositivos.
- Según su función en la aplicación de usuario:
 - Dispositivo de funcionalidad completa: También conocido como nodo activo. Es capaz de recibir mensajes en formato 802.15.4. Gracias a la memoria adicional y a la capacidad de computar, puede funcionar como coordinador o enrutador ZigBee. También puede ser usado en nodos que actúen de interfaz con los usuarios.
 - Dispositivo de funcionalidad reducida: También conocido como nodo pasivo. Tiene capacidad y funcionalidad limitadas (especificada en el estándar) con el objetivo de conseguir un bajo coste y una gran simplicidad. Básicamente, son los sensores/actuadores de la red.

Los dispositivos ZigBee fueron pensados para aplicaciones con una baja tasa de transferencia y en ráfagas eventuales. Esto permite estar en modo bajo consumo (dormido) la mayor parte del tiempo ahorrando vida útil de la batería. Cuando un nodo es requerido es capaz de despertarse en un tiempo ínfimo (15ms aproximadamente). Luego de ser usado puede volver rápidamente a estado dormido y seguir ahorrando batería.

Estas características de funcionamiento hacen que ZigBee trabaje en márgenes muy estrechos de consumo y ancho de banda. Por eso, para la certificación de los dispositivos, es necesaria una validación completa de los requerimientos a nivel físico para garantizar la funcionalidad del hardware. Esto está especificado en el estándar 802.15.4, donde se indica que los radios deben pasar validaciones ISO 17025. [10]

Ventajas:

- Se pueden conectar hasta 65000 nodos en una red.
- Soporta diferentes topologías.
- La topología en malla proporciona confiabilidad a la red ante eventuales fallas, pérdidas de señal o de dispositivos. Seguirá funcionando ya que se establecerán nuevos caminos de comunicación.
- La baja tasa de transferencia y ciclo de trabajo se adaptan perfectamente a aplicaciones de domótica donde se logrará una optimización de la vida útil de las baterías.

Desventajas:

- La baja tasa de transferencia puede ser inadecuada para cierto tipo de aplicaciones.
- Un dispositivo por sí mismo tiene poco alcance, esto puede no adecuarse a aplicaciones con poca cantidad de dispositivos.
- Si bien el hardware es sencillo, las rigurosas certificaciones que se deben validar a nivel físico hacen que sea más costoso el precio por dispositivo. Además, la masificación que han alcanzado otros estándares como WiFi y Bluetooth abarató la producción de estos, aumentando la brecha de costos con ZigBee.

1.3.3.3 WiFi

Es un mecanismo de conexión de dispositivos electrónicos orientado a la formación de redes locales inalámbricas WLAN. Se encuentra estandarizado bajo la norma IEEE 802.11 la cual es aplicada para garantizar compatibilidad entre distintos fabricantes. [11]

Al igual que otras normas 802 para redes locales de IEEE, WiFi también se centra en las capas inferiores del modelo OSI, el medio físico y el enlace. Esto implica total compatibilidad con Ethernet y permite a los usuarios correr cualquier protocolo o aplicación de capas superiores sin importar el medio físico y enlace utilizado, por ejemplo TCP y UDP entre las más importantes.

Desde su creación, el estándar ha ido actualizándose y mejorando. A continuación se muestra en una tabla la comparativa entre las versiones de uso masivo y las mejoras que estas implementaron.

Norma	Año	Tasa Teórica	Tasa Real	Frecuencia	Características
802.11b	1999	11Mbps	5,9-7,1Mbps	2,4GHz	Método de acceso al medio CSMA/CA primera norma con alcance masivo
802.11g	2003	54Mbps	22Mbps	2,4GHz	Compatible con 802.11b, pero deteriora la tasa de transferencia
802.11n	2009	600Mbps	80-100Mbps	2,4GHz y 5GHz	MIMO permite usar varios canales en simultaneo y aumentar el alcance
802.11ac	2014	1,3Gbps	433Mbps	5GHz	Opera en canales menos interferencias, menor alcance que en la banda ISM

Tabla 1.6 Versiones de normas 802.11

Existen dos tipos de hardware diferente que intervienen en la creación de una red WiFi, las estaciones (Station) y los puntos de acceso (Access Point).

El Acces Point o AP brinda los parámetros bajo los cuales se establece la red WiFi, el canal de comunicación, la seguridad, etc. y posee una identificación para diferenciarlo de otras redes inalámbricas que puedan existir en el área.

La estación tiene la capacidad de escanear las redes disponibles, de elegir y unirse a cualquiera de ellas siempre y cuando cumpla con los requisitos. Para ello, cada AP envía una trama con la información de la red a intervalos regulares. Estos paquetes son recolectados por las estaciones y de esta forma las estaciones “conocen” los AP disponibles. Si la estación desea unirse a una red particular envía una solicitud de sondeo con los datos necesarios para unirse los cuales son procesados por el AP el cual puede aceptar o rechazar a la estación según corresponda. Esta forma de funcionamiento es denominada infraestructura, ya que requiere de un AP para la conformación de la red. El AP es el que proporciona el servicio y las estaciones son clientes.

Existe otro modo de funcionamiento el cual no es requerido un AP y es el modo AD-HOC. En este modo, los clientes de la red inalámbrica se conectan entre sí para formar una red punto a punto, de esta forma actúan como estación y como AP en simultaneo. En AD-HOC, si una estación quiere mandar información a otra que esta fuera de rango, no podrá hacerlo a pesar de estar al alcance de otras estaciones de la misma red. Esto es debido a la naturaleza punto a punto de la comunicación.

En el modo infraestructura, la red posee una arquitectura centralizada, mientras que en AD-HOC es distribuida. Las ventajas y desventajas de una arquitectura centralizada y distribuida ya se han analizado con anterioridad, vale mencionar que en este caso, normalmente una red con infraestructura posee más alcance que una red AD-HOC.

La seguridad en este tipo de red es un aspecto primordial. Se han diseñado varios sistemas de seguridad para impedir que equipos no autorizados ingresen a la red y para proteger los datos que se transmiten. Los mecanismos más conocidos son:

WEP: Protege el acceso a la red mediante una contraseña, pero la información que viaja no está protegida.

WPA: Protege el acceso a la red mediante una contraseña a la vez que encripta la información que se transmite.

Filtrado por MAC: Mediante esta característica, el administrador de la red puede predefinir los equipos que están autorizados a asociarse a la red. Los equipos que no estén registrados se rechazarán a pesar de tener la clave de conexión correcta. Esta protección no encripta los datos al transmitirlos.

Ventajas:

- Al tratarse de una WLAN, ofrece el mejor alcance entre las otras tecnologías de redes inalámbricas vistas anteriormente, las cuales son WPAN.
- Soporta gran cantidad de dispositivos.
- Es compatible con las aplicaciones más importantes de capas superiores, como lo son IP, TCP y UDP entre otras, permitiendo lograr redes más complejas y con mayores prestaciones.
- Es compatible y se puede integrar a redes Ethernet.
- Puede coexistir con enlaces Bluetooth debido a los métodos de salto en frecuencia que utilizan.
- Permite grandes tasas de transferencia de datos.
- Al ser una tecnología de amplia utilización que ha logrado penetrar en el mercado el costo es bajo.
- Garantiza compatibilidad entre fabricantes.

Desventajas:

- El alcance es afectado por el entorno.
- A excepción de las últimas actualizaciones, utiliza la banda ISM a 2,4GHz la cual se encuentra saturada y la presencia de otros equipos que trabajen en esa frecuencia puede alterar el funcionamiento de la red.
- El consumo de energía de los dispositivos que utilizan esta tecnología es elevado en comparación a otros sistemas de redes de datos inalámbricos.
- Debido a la sensibilidad de los datos que circulan por este tipo de red y la cantidad de servicios que permite, la seguridad de la misma se convierte en un aspecto crítico. El sistema de seguridad WEP posee serias vulnerabilidades que se compensaron con el sistema WPA. También el filtrado por MAC y el ocultamiento de la red son herramientas igualmente importantes para garantizar la seguridad.

1.3.4 Interfaz de usuario

La interfaz de usuario es el mecanismo que permite al usuario operar el sistema domótico. Existen una infinidad de posibilidades y combinaciones, se busca maximizar la comodidad y la accesibilidad para todo tipo de persona.

Los métodos de ingreso más utilizados suelen ser terminales con teclas fijas estratégicamente ubicadas o portátiles, acceso web y reconocimiento de voz. También se suelen usar tableros con LED, display LCD, locuciones pregrabadas y páginas web para que el usuario pueda obtener una devolución del estado actual sistema.

La implementación de la interfaz puede realizarse directamente en el diseño en un dispositivo dedicado a este uso exclusivamente o puede implementarse por medio de una aplicación que corra en un sistema operativo ya sea de PC, aplicación web, *smartphones*, *tablets*, etc.

Uno de los objetivos de este proyecto es realizar una interfaz de manos libres, para lo cual se determinó la implementación con un sistema reconocedor de voz, por ser la opción más versátil. A continuación se analizarán algunas opciones que brindan soluciones integradas para este propósito. Se eligen estas opciones por ser las que poseen un precio más competitivo en el mercado.

1.3.4.1 Modulo VR de Elechouse

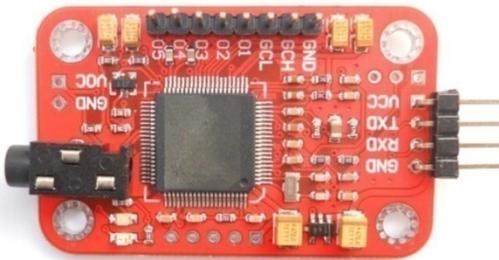


Figura 1.15 Modulo VR Elechouse

Este módulo, desarrollado por Elechouse, permite reconocer la voz. Puede almacenar hasta 80 comandos de voz. Estos 80 comandos están divididos en grupos de hasta 7 comandos cada uno. Primero se debe entrenar el módulo con las instrucciones, grupo por grupo. Antes de reconocer algún comando debe elegirse el grupo. Al establecerse un grupo reconocerá los comandos que estén almacenados en él. Si se dice algún comando que no está registrado o está registrado en otro grupo resultara en un error. Para reconocer un comando que está en otro grupo se debe primero establecer el grupo y luego proceder con el reconocimiento. Recibe la configuración por medio de comandos a través de una interfaz de puerto serie. Este modulo es dependiente del locutor. [12]

Ventajas:

- Muy Fácil implementación.
- Compatible con cualquier dispositivo que posea interfaz UART.
- Bajo costo.

Desventajas:

- Solo posee 80 comandos divididos en grupos 7 comandos cada uno.
- Al ser dependiente del locutor, cada comando solo será reconocido por la persona que lo entrenó.

1.3.4.2 SimpleVR de Elechome

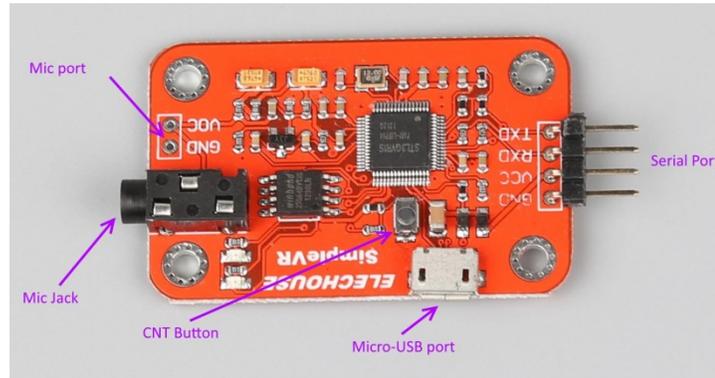


Figura 1.16 Modulo SimpleVR

A diferencia de su predecesor descrito en el apartado anterior, este modulo es independiente del locutor. Por lo tanto permite que los comandos sean reconocidos de cualquier persona. No necesita ser entrenado, pero en su lugar, para cargarle los comandos es necesario conectarlo a una computadora. Mediante un software especial se introducen en texto los comandos que luego serán procesados para que los reconozca del audio que se toma por la entrada de micrófono. [13]

El resto del funcionamiento es igual, toma las instrucciones mediante puerto serie. Mediante estas instrucciones se puede establecer el grupo del cual se quiere reconocer el comando y disparar la orden de reconocer (oír) un comando.

En esta versión las capacidades de memoria fueron incrementadas significativamente. Ahora el dispositivo posee 64 grupos de 2000 comandos cada uno.

El sistema es independiente del locutor, pero no del idioma del locutor. Solo reconoce comandos en Ingles y en Mandarín.

Ventajas:

- Muy Fácil implementación.
- Compatible con cualquier dispositivo que posea interfaz UART.
- Posee gran capacidad de memoria para guardar comandos.
- Reconoce comandos de voz independientes del locutor.
- Bajo costo.

Desventaja:

- No es compatible con el idioma español.

1.3.4.3 EasyVR de Veeear



Figura 1.17 Modulo EasyVR

EasyVR se trata de un modulo multipropósito diseñado para integrar de forma sencilla, versátil y a bajo costo capacidades de reconocimiento de voz a casi cualquier aplicación. [14]

Posee una interfaz UART que permite ser comandado y conectarse con otros sistemas dentro de la aplicación. A su vez posee 6 salidas de propósitos generales que pueden ser controlados mediante comandos UART.

El modulo contiene un conjunto de comandos predefinidos de uso común, que son independientes del locutor. En total son 26 comandos divididos entre 4 grupos. Y se encuentran disponibles en inglés, italiano, japonés, alemán, español y francés.

Posee hasta 32 comandos dependientes del locutor definidos por el usuario, para los cuales es necesario realizar un entrenamiento del dispositivo. Dichos comandos se almacenan en grupos.

También posee la capacidad de grabar hasta 28 comandos personalizados independientes del locutor, también compatibles con los mismos idiomas, con un máximo de 12 por grupo. Para subirlos al modulo es necesario un ordenador y un software que provee el fabricante mediante una licencia.

La operación para el reconocimiento es similar a los dispositivos antes mencionados. Mediante comandos por puerto serie se indica que se escuche determinado grupo de comandos (que pueden ser independientes o dependientes del locutor). El dispositivo escuchara por un determinado tiempo y luego dará cuenta del resultado de la operación mediante una respuesta por puerto serie.

Este dispositivo también cuenta con una salida de audio que permite reproducir sonidos y tonos DTMF por ejemplo.

El fabricante garantiza una serie de herramientas de software para brindar al proyectista facilidades para integrar el modulo en la aplicación que se esté desarrollando. Una de las más destacables es una librería para la plataforma de desarrollo Arduino.

Ventajas:

- Procesa comandos tanto dependientes como independientes del locutor.
- Trae integrado un conjunto de comandos independientes del locutor pregrabados de uso general que permiten una rápida implementación del modulo en cualquier proyecto.
- Se pueden generar comandos independientes del locutor personalizados.
- El fabricante pone a disposición librerías que facilitan el desarrollo de proyectos.
- Al poseer una relativamente buena capacidad de almacenamiento permite guardar una cantidad suficiente para casi cualquier uso que se quiera aplicar. Esto es incluyendo tanto los comandos independientes como los dependientes del locutor.
- Es compatible con varios idiomas, incluyendo el español.

Desventaja:

- El costo de este modulo es el mayor entre los otros dispositivos analizados en este apartado. Para implementar comandos independientes del locutor es necesario adquirir un software con licencia.

1.3.4.4 Google Speech Cloud API

Se trata de una interfaz de aplicación (API) online que provee Google para el reconocimiento de voz. Utiliza el mismo motor que las aplicaciones de Google. Su eficiencia y robustez está altamente comprobada por la mayoría de los usuarios que hacen uso de esta tecnología en internet. Se ha probado su uso efectivo en muchos idiomas y en condiciones ambientales con diferentes tipos de ruido. [15]

El nivel de desarrollo que posee esta aplicación es muy elevado, digno de una multinacional como lo es Google. Por lo tanto no necesita ningún entrenamiento ni programación de ningún dispositivo previo, solo la conexión adecuada a internet para poder acceder al servicio Web. El servicio transforma a texto el discurso que es enviado para procesar.

Ventajas:

- Es uno de los sistemas más robustos de reconocimiento de voz.
- No necesita ningún entrenamiento previo.
- Como el servicio consiste en transformar audio a texto, no tiene un límite en el vocabulario que utiliza para identificar comandos.
- El costo del servicio es por el tiempo del audio que es procesado y luego transcrito y es relativamente accesible.

Desventaja:

- Es un servicio online, requiere de conexión a internet para que funcione.

1.3.5 Sensores y actuadores

Los sensores son los encargados de recolectar información de variables de interés y transformarla en variables eléctricas que el sistema pueda manipular y procesar. Los sensores más comunes utilizados son los de humedad, temperatura, humo, presencia, luminosidad, posición, etc.

Existen otros parámetros importantes de un sensor además del tipo de variable que mide. Se deben tener en cuenta el rango de temperatura de operación, el rango de tensión de salida que entrega, la tensión de alimentación, la potencia de consumo y la precisión de medición entre otros.

Los actuadores reciben órdenes del controlador para concretar acciones. Los más comunes son los relés de actuación, las electroválvulas, sirenas, parlantes, motores. También es importante evaluar las condiciones de operación dentro de un entorno doméstico, las tensiones de alimentación, potencia de consumo, temperatura de operación, etc.

1.4 Conformación del sistema

En esta Sección se evaluarán las diferentes opciones para conformar cada parte del sistema domótico. Se elegirán algunas de las tecnologías, protocolos y arquitecturas descritas en el apartado anterior. Para ello se aplicarán criterios basados en los objetivos que persigue este proyecto.

1.4.1 Elección de Interfaz de Usuario

De las soluciones disponibles para el reconocimiento de voz el principal factor que se tuvo en cuenta es que pudiera manejar una buena cantidad de comandos para facilitar su uso. En el proyecto se considero prioritaria la posibilidad de ir agregando cada vez más dispositivos periféricos, lo cual implica un menú cuyos elementos navegables van en aumento. Si se contaran con pocos comandos de voz, esto actuaría como una limitación para la cantidad de opciones que se pueden navegar en el menú. Si bien el menú se podría paginar, depende en la medida que fuera necesario, puede tornarse un inconveniente para la facilidad de uso.

Por otro lado, se tuvo en cuenta la funcionalidad dependiente/independiente del locutor ya que tiene un impacto en la practicidad del uso y en la facilidad de instalación. Si los comandos fueran dependientes del locutor, antes de poder usar el sistema sería necesario entrenar el dispositivo, lo cual conduce a un esfuerzo de parte del usuario final. Además el sistema sería solo utilizable por una persona. Si bien se pueden utilizar para una personalización del uso, se buscó que no sea necesaria esa configuración inicial y que también permita usarse por diferentes personas. Por estas razones, el modulo VR de Elechouse quedó descartado, ya que posee 80 comandos, todos dependientes del locutor y no alcanza a las especificaciones de este proyecto.

El segundo módulo de Elechouse SimpleVR, posee más cantidad de comandos y son independientes del locutor. Este tipo de comandos son sensibles al idioma, es decir, que identifican las palabras de cualquier locutor siempre y cuando esté hablando en el idioma que fue predefinido mediante la configuración. Como SimpleVR solo es compatible con el idioma inglés y el mandarín no se implementó esta solución ya que en este proyecto se apunta a usuarios que hablan castellano.

El servicio web de Google es independiente del locutor, funciona en castellano y al ser online no tiene límite de memoria. Como transforma en texto la voz, se podría usar un microcontrolador para procesar dicho texto e interpretar la orden. Como este proyecto apunta a aumentar la independencia del usuario, se considero que al necesitar de internet (y por lo tanto de prestadores terceros de servicio, tanto de Google como el proveedor de internet) el sistema perdería confiabilidad. De esta forma el usuario podría tener interrupciones del servicio impidiendo el uso del sistema. Por eso no se implementó esta opción.

El módulo EasyVR de Veeear es el modulo que se utilizó en el proyecto y esto es debido a la versatilidad que posee en cuanto a cantidad y tipo de comandos. Además puede procesar palabras independientes del locutor sin depender de internet. Este tipo de comandos se incluyen pregrabados en el dispositivo y poseen palabras de uso general en varios idiomas, incluyendo español. Esto permite que se pueda comenzar a utilizar el sistema sin ningún entrenamiento previo. También posee la capacidad de entrenar comandos dependientes del locutor en caso que el usuario desee personalizar el uso del equipo, lo cual cuenta como un valor agregado. Si fuera necesario, es posible integrar comandos independientes del locutor personalizados. Esto siempre y cuando se cuente con la licencia de un software especial, lo cual puede tenerse en cuenta para futuras mejoras del proyecto.

Esta elección condujo a la determinación de una plataforma para implementar el resto de la interfaz de usuario, ya que los resultados del reconocimiento de voz controlan la navegación de un menú.

EasyVR se comunica por puerto serie, desde ahí se configura el módulo para su uso y se toman los resultados del reconocimiento, por lo tanto es compatible con cualquier plataforma que maneje USART, como Raspberry o PIC. Sin embargo se optó por la plataforma Arduino ya que es la única plataforma que el fabricante del modulo facilita las librerías para su programación.

Se decidió utilizar un display LCD para la visualización del menú, de forma de aprovechar los recursos que provee la placa Arduino. Así mismo también sirvió para la implementación de la gestión de datos de los dispositivos y control de la interfaz de comunicación.

1.4.2 Elección de la Infraestructura de Comunicación

Siguiendo el criterio de brindar una instalación del sistema lo más sencilla posible, se descartaron las opciones de comunicación cableadas. En esta sección discutirán las alternativas inalámbricas que se tuvieron en cuenta para realizar la infraestructura de comunicación.

En términos tecnológicos, ZigBee es la opción más adecuada, si bien posee bajo ancho de banda, es suficiente para las aplicaciones de domótica. También maximiza el uso de baterías y al usar topología en malla la comunicación es robusta. Pero el costo con respecto a Bluetooth y WiFi es superior, por eso se descartó esta alternativa.

Los módulos Bluetooth tienen costos similares a los de WiFi, esto es debido a la escala a la cual son producidas estas tecnologías. Bluetooth tiene un costo energético menor a WiFi, pero también el alcance en metros del enlace es inferior.

En comparación a WiFi, Bluetooth ofrece una cantidad de dispositivos conectados en simultáneo más limitada y el rendimiento de la comunicación es afectado por la cantidad de dispositivos que conforman la red. Por esta razón se decidió utilizar módulos WiFi, ya que de esta forma la cantidad de dispositivos que se pueden conectar en simultáneo son de aproximadamente 250, dependiendo de los dispositivos que el usuario ya tenga conectados en la red (esto en el caso de ser compartida con otros equipos).

Se consideraron soluciones que permitan conectarse a WiFi de forma embebida. Se eligió el módulo ESP8266 de Espressif el ya que posee un costo muy competitivo.

El chip contiene en el mismo encapsulado los periféricos necesarios para conectarse a WiFi e integra de forma completa la pila TCP/IP. También posee entradas y salidas de propósito general, salidas PWM, entradas a conversores analógico digital, timers, entre otras. Se comercializa en módulos donde el chip viene insertado en un circuito impreso. Estos vienen en diferentes tamaños y formas de acuerdo al modelo, con más o menos entradas disponibles y con capacidad de memoria flash variable, ya que el chip no posee memoria integrada.

Espressif pone a disposición de la comunidad diferentes SDK's (*Software Development Kits*). Estos han permitido desarrollar una gran cantidad de herramientas por parte de los propios usuarios, las cuales se contribuyen y permiten seguir escalando las prestaciones que ofrece esta plataforma. Se trata de una tecnología en pleno auge y la cual promete un futuro prolífico.

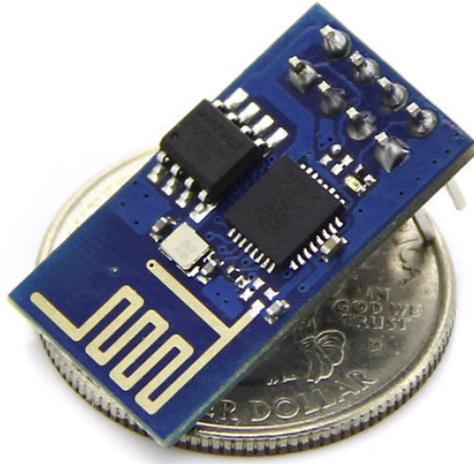


Figura 1.18 Modulo ESP8266 (Modelo ESP-01)

Estos módulos funcionan en modo AP (*Access Point* o Punto de Acceso) y *Station* (Estación) en simultáneo. Aunque en modo AP no tolera más de 4 dispositivos conectados en total. Esto es porque fueron diseñados para redes en malla. Sin embargo, si existe un AP externo, como puede ser un enrutador, se pueden conectar tantos módulos como este AP anfitrión permita. Por eso, para este proyecto es necesaria una red WiFi anfitriona.

Estos dispositivos fueron diseñados con un enfoque IoT. Si bien están listos para conectarse a internet y son compatibles con aplicaciones en la nube, no depende de estas para su función en el proyecto. Sin embargo esta capacidad de conexión a internet se puede aprovechar para aumentar las capacidades del sistema una ventaja a la hora de presentar el producto al mercado.

1.4.3 Elección del Protocolo

Los protocolos X-10 y Konnex han sentado precedentes en la domótica y hoy en día siguen siendo tan vigentes como en el momento de su creación. Para este proyecto, que se decidió implementar de forma inalámbrica, el protocolo X-10 queda obsoleto. Sin embargo su sencillez en cuanto a la instalación y su estilo modular orientado a la expansión es algo que se tomó como modelo a seguir en este proyecto. En cuanto al protocolo Konnex, si bien es compatible con medios de transmisión inalámbricos, no se adapta fácilmente al hardware que se utilizó en este proyecto. A su vez, sus mayores ventajas se encuentran en proyectos de mayor envergadura y no se justificó su aplicación para este proyecto particular.

Aprovechando las funcionalidades que proveen los módulos ESP8266, se ha decidido dar un enfoque IoT a este proyecto. Se consideró que utilizar este enfoque facilitará la inserción del producto en el mercado ya que se trata de una tecnología de vanguardia. Al ser un área en pleno crecimiento, proporciona una plataforma que se puede ir actualizando permitiendo la introducción constante de nuevas características.

Lo que se incorpora del concepto IoT, es la direccionabilidad de los objetos y su codificación. Esto permite la transformación de su interacción con el usuario. No se considera necesaria la conectividad a internet, ya que el sistema debe prestar la funcionalidad establecida como objetivo de forma offline. Así se puede garantizar una experiencia de uso confiable. Bajo este esquema, el objeto funciona por sí mismo y su interfaz estará disponible de forma independiente siempre y cuando esté conectado a la red WiFi. No

requiere de una central que lo coordine. Esto marca un lineamiento en la topología de control que tendrá el sistema, lo cual se detallará en el siguiente apartado.

La conectividad a internet del sistema es optativa y expande las capacidades de uso, permitiendo monitorización y control a distancia. Dicha característica cobra mucha importancia si consideramos la alta disponibilidad de los dispositivos móviles con acceso a internet. Esto puede resultar atractivo para todos los usuarios por igual, los que poseen alguna discapacidad motriz y los que no. De esta forma queda el camino preparado para futuras mejoras en el sistema, que se pueden lograr de forma muy sencilla. Al momento de desarrollar este prototipo esto excedía los requisitos del proyecto, pero se puede agregar como característica para introducir el sistema como un producto en el mercado.

1.4.4 Elección de Topología

Los módulos ESP8266 están dotados de conectividad WiFi y cuentan con toda la pila TCP para comunicarse. Esto le permite a cada uno (y por ende cada periférico que cuenta con él) ser independiente para conectarse a la red, enviar o recibir mensajes. Si se sigue el esquema del modelo en capas TPC/IP [16], la comunicación no depende de la coordinación de ningún otro elemento que forme parte del sistema domótico, sino que queda relegada a una capa inferior en el modelo. El sistema domótico estaría en la capa de aplicación en el modelo TCP/IP, abstraído de lo que acontece en capas inferiores.

Cada elemento que formara parte del sistema, ya sea un periférico (que puede ser un actuador o un sensor) como una interfaz de usuario (que podría haber una o varias), se podría comunicar con otros pares siempre y cuando estén conectados a una misma red WiFi y conozcan a qué dirección IP a la cual deban comunicarse. Si fallara algún dispositivo, el resto seguirá funcionando de acuerdo a su programación. Incluso los periféricos seguirán operando si no hay ninguna interfaz de usuario conectada y se comunicaran entre ellos si es requerido. Por lo tanto, desde un punto de vista del sistema domótico, la inteligencia se encuentra distribuida. La infraestructura WiFi hace las veces de un BUS de comunicación, si el bus cae, también lo hará la comunicación entre todos los dispositivos y tendrá que ser reemplazado para restaurar el vínculo.

1.4.5 Elección de Periféricos

Con el fin de reducir costos, se ha decidido usar los recursos del modulo ESP8266 para implementar los periféricos. Además de su funcionamiento básico como interfaz de comunicación WiFi, este chip provee capacidades de programación. Esto permitió implementar rutinas que aportaon la inteligencia para lograr la aplicación de domótica. A modo de prueba del prototipo se construyeron dos dispositivos periféricos. Cada uno está constituido por una placa ESP8266. Ambos son de dos estados, encendido y apagado. Poseen una luz LED que se puede activar desde la interfaz de usuario. Estos dispositivos son de baja potencia. Para que puedan actuar sobre otros elementos de mayor potencia, como podría ser una lámpara o un motor, debería agregarse una etapa adecuada para operar un relay o algún elemento similar.

También, estos prototipos de periféricos se podrían utilizar como sensores de dos estados, para luego mostrar la lectura en la interfaz de usuario. Para ello basta cambiar la programación del microcontrolador que llevan integrado.

La placa ESP8266 también permite realizar un periférico multiestado. Para realizar un actuador, se pueden emplear dos opciones, el puerto serie y las salidas PWM. En ambos casos se necesita una etapa adicional para acondicionar la señal al dispositivo que se desea manipular. En el caso de querer realizar un sensor, también se puede usar el puerto serie o una entrada a un conversor analógico digital con la

circuitería adicional para adaptar la señal del sensor que se desea monitorear. Dependiendo de la placa ESP utilizada es posible que el periférico sea sensor y actuador en simultáneo, siempre y cuando tenga las entradas disponibles para dicha aplicación. No todas las placas ESP tienen los mismos pines disponibles.

A pesar de que los periféricos prototipos son biestables, la interfaz de usuario se diseñó para poder leer datos de periféricos multiestado.

1.5 Conclusiones

El sistema propuesto está compuesto por una interfaz de usuario y dos actuadores periféricos. La comunicación entre los diferentes elementos se lleva a cabo mediante una infraestructura WiFi, para lo cual se usan microcontroladores con conectividad WiFi embebida y pila TCP/IP completa.

La interfaz de usuario se controla mediante identificación de voz, para ello se emplea el módulo EasyVR que cuenta con comandos independientes del locutor en español y con comandos dependientes del locutor entrenables. Para visualizar el estado del sistema se usa un menú en una pantalla LCD cuyo contenido es dinámico. Para comunicarse con los periféricos se conecta a un módulo ESP8266 a través de un puerto serie.

La interfaz de usuario se encarga de las siguientes tareas:

- Control del LCD.
- Control del menú.
- Control de módulo VR.
- Prepara órdenes para los periféricos.
- Recolecta información de los periféricos.
- Envía órdenes a través de interfaz WiFi por ESP8266.

Todas estas funciones las coordina una placa Arduino.

Los dispositivos periféricos realizan las siguientes funciones:

- Procesan los mensajes recibidos y detectan comandos.
- Envían mensajes informando su estado.
- De acuerdo a las órdenes recibidas encienden o apagan una luz LED.

Todas estas funciones las procesa una placa ESP.

Como la inteligencia del sistema domótico quedó implementada en la capa de aplicación del modelo TCP/IP, la topología de comunicación de ese nivel es distribuida. No existe ninguna central o agente coordinador que sea requerido para mantener la comunicación. Pueden quedar desconectados de la red varios elementos del sistema domótico, pero los que quedan conectados se pueden comunicar entre sí en el caso de ser requerido. La comunicación es dependiente de la infraestructura WiFi, en el caso de fallar se deberá buscar otra red anfitriona para que el sistema funcione.

El acceso a la infraestructura WiFi quedó relegado a la placa ESP, por lo tanto, en la interfaz de usuario, la placa Arduino no interviene en el proceso, esto se detallará en el siguiente Capítulo.

En la Figura 1.19 se presenta un diagrama en bloques de los componentes del sistema propuesto.

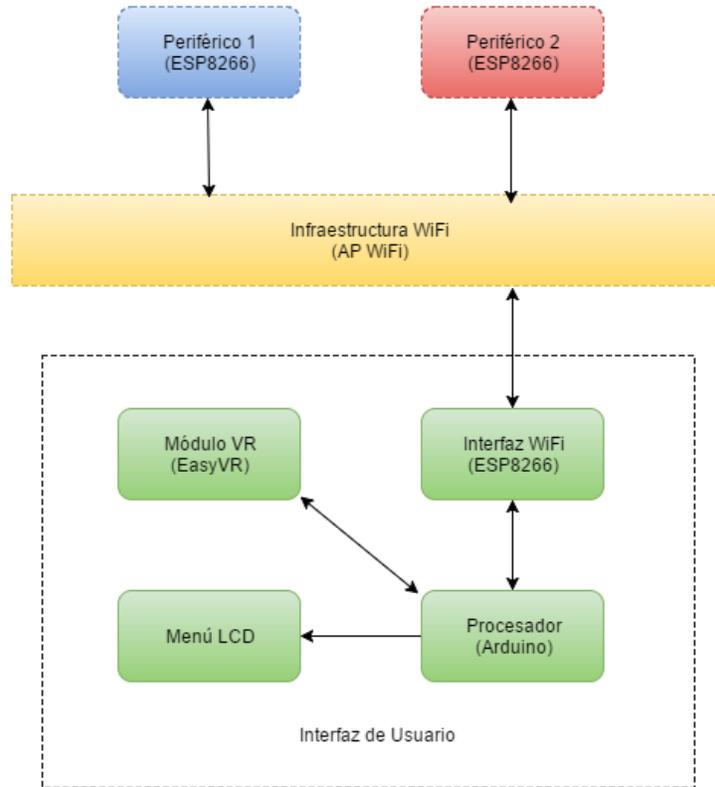


Figura 1.19 Diagrama en bloques del sistema propuesto

CAPÍTULO 2

2.1 Introducción

En este Capítulo se estudiarán las características que posee el módulo WiFi ESP8266. Este se utilizó como interfaz WiFi en todos los dispositivos que integran el sistema. De esta forma se les proporcionó la capacidad de conectarse a una red WiFi estableciendo un medio de comunicación entre estos.

Luego se discutirán los problemas y soluciones que surgieron al querer establecer un método sencillo para asociar los dispositivos a la red WiFi anfitriona.

Finalmente se desarrolla un mecanismo de referencia para que los equipos puedan hallar a sus pares una vez conectados a la red WiFi.

2.2 Características del módulo ESP8266

ESP8266EX es un SoC (*System on Chip*) altamente integrado diseñado para satisfacer las constantes demandas de uso eficiente de energía, diseño compacto y robustez en la industria de la “internet de las cosas” (IoT por sus siglas en inglés). [17]

Con las capacidades WiFi totalmente autocontenidas, el ESP8266 puede ejecutar aplicaciones por su propia cuenta como también aplicaciones donde responde a la coordinación de algún microcontrolador. Cuando el ESP8266EX alberga la aplicación, inicia rápidamente desde la memoria flash. El cache integrado de alta velocidad ayuda a incrementar el rendimiento y optimiza la memoria del sistema. También, el ESP8266EX puede ser aplicado a cualquier diseño con microcontroladores como un adaptador WiFi a través de las interfaces SPI / SDIO ó I2C /UART.

ESP8266EX integra interruptores de antena, balún RF, amplificador de potencia, amplificador de bajo ruido en receptor, filtros y módulos administradores de energía. El diseño compacto minimiza el espacio en circuito impreso (PCB por sus siglas en inglés) y requiere mínimos circuitos externos.

Además de las funcionalidades WiFi, el chip integra una versión mejorada del procesador de 32 bits de Tensilica, L106 Diamond y memoria RAM. Este puede ser conectado con sensores externos y otros dispositivos por medio de las entradas y salidas de propósitos generales (GPIO).

El chip posee sofisticadas características, como el rápido traspaso de modo bajo consumo y modo despierto para propósitos de eficiencia energética, polarización de radio adaptativa para operación en bajo consumo, procesamiento de señales avanzado, cancelación de espurias y mecanismos de coexistencia de la señal RF con otros sistemas para mitigar interferencias.

2.2.1 Protocolos WiFi

- Soporta 802.11 b/g/n/e/i.
- Soporta WiFi Direct (P2P).
- Descubrimiento P2P, modo P2P Group Owner, P2P Power Management.
- Modo estación de infraestructura BSS / Modo P2P / Modo softAP.
- Aceleraciones de hardware para CCMP (CBC-MAC, counter mode), TKIP (MIC, RC4), WAPI (SMS4), WEP (RC4), CRC.
- WPA/WPA2 PSK, y WPS driver.
- Características de seguridad adicionales 802.11i como pre-autenticación, y TSN.

- Interfaz abierta para varios esquemas de autenticación de capas superiores sobre EAP como TLS, PEAP, LEAP, SIM, AKA.
- Soporte 802.11n (2.4 GHz).
- Soporta MIMO 1x1 y 2x1, STBC, A-MPDU y agregado de A-MSDU e intervalo de guarda de 0.4µs.
- WMM *power save* U-APSD.
- Administración de múltiples colas para utilizar totalmente la priorización del tráfico como es definido por el estándar 802.11e.
- UMA compatible y certificado.
- Encapsulamiento de trama 802.1h/RFC1042.
- DMA dispersado para un óptimo *CPU off load* en operaciones *Zero Copy* de transferencia de datos.
- Diversidad y selección de antena (hardware administrado por software).
- *Clock/power gating* combinado con administración de energía compatible con 802.11 adaptado a la condición de conexión de corriente proveyendo mínimo consumo de energía.
- Algoritmo adaptativo de retroceso de velocidad para establecer la tasa de transmisión y consumo de energía óptimos basados en el SNR real y la información de paquetes perdidos.
- Respuesta y retransmisión automática en MAC para evitar descarte de paquetes en situaciones de anfitrión lento.
- Soporta *roaming* sin interrupción.
- Arbitraje de tráfico de paquetes configurable (PTA) con procesador esclavo dedicado basado en diseño, provee un flexible y exacto *timing* para dar soporte a la coexistencia con Bluetooth para un amplio rango de fabricantes de Chips Bluetooth.
- Soporte para coexistencia de antena Bluetooth doble y simple con recepción simultánea óptima (WiFi/Bluetooth).

2.2.2 Especificaciones Técnicas

Categorías	Ítems	Parámetros	
WiFi	Standards	FCC/CE/TELEC/SRRC	
	Protocolos	802.11 b/g/n/e/i	
	Rangos de Frecuencia	2.4 G ~ 2.5 G (2400M ~ 2483.5M)	
	Potencia de transmisión		802.11 b: +20 dBm
			802.11 g: +17 dBm
			802.11 n: +14 dBm
	Sensibilidad de recepción		802.11 b: -91 dBm (11 Mbps)
			802.11 g: -75 dBm (54 Mbps)
			802.11 n: -72 dBm (MCS7)
	Antena	Trazo PCB, Externa, Conector IPEX, Chip Cerámico	

Tabla 2.1 Especificaciones de WiFi

Categorías	Ítems	Parámetros
Hardware	CPU	Tensilica L106 32-bit micro controller
	Interfaces	UART/SDIO/SPI/I2C/I2S/IR Control Remoto
		GPIO/ADC/PWM
	Tensión de trabajo	3.0 V ~ 3.6 V
	Corriente de operación	Valor promedio: 80 mA
	Rango de temperatura de operación	-40°C ~ 125°C
	Rango de temperatura de almacenamiento	-40°C ~ 125°C
	Tamaño del encapsulado	QFN32-pin (5 mm x 5 mm)
Interfaz externa	-	

Tabla 2.2 Especificaciones de Hardware

Categorías	Ítems	Parámetros
Software	Modo Wi-Fi	station/softAP/softAP+station
	Seguridad	WPA/WPA2
	Encriptación	WEP/TKIP/AES
	Actualización del firmware	UART Download / OTA (via network)
	Desarrollo de Software	Soporta firmwares de desarrollo de servidores en la nube y SDK para rápida programación <i>on-chip</i>
	Protocolos de red	IPv4, TCP/UDP/HTTP/FTP
	Configuración	Comandos AT, Servidores en la nube, Android/iOS App

Tabla 2.3 Especificaciones de Software

2.2.3 Características Eléctricas

Parámetros	Condiciones	Mín	Típica	Máx	Unidad	
Rango de temperatura de almacenamiento	-	-40	Normal	125	°C	
Temperatura máxima de soldadura	IPC/JEDEC JSTD-020	-	-	260	°C	
Valor de tensión de trabajo	-	3.0	3.3	3.6	V	
I/O	VIL/VIH	-	-0.3/0.75VIO	-	0.25VIO/3.6	V
	VIL/VIH	-	N/0.8VIO	-	0.1VIO/N	V
	IMAX	-	-	-	12	mA
Descarga electrostática (HBM)	TAMB=25°C	-	-	2	KV	
Descarga electrostática (CDM)	TAMB=25°C	-	-	0.5	KV	

Tabla 2.4 Características eléctricas

2.2.4 Consumo Energético

Parámetros	Mín	Típica	Máx	Unidad
Tx802.11b, CCK 11Mbps, P OUT=+17dBm	-	170	-	mA
Tx 802.11g, OFDM 54Mbps, P OUT =+15dBm	-	140	-	mA
Tx 802.11n, MCS7, P OUT =+13dBm	-	120	-	mA
Rx 802.11b, tamaño de paquete 1024 bytes, -80dBm	-	50	-	mA
Rx 802.11g, tamaño de paquete 1024 bytes, -70dBm	-	56	-	mA
Rx 802.11n,tamaño de paquete 1024 bytes, -65dBm	-	56	-	mA
Modem-Sleep	-	15	-	mA
Light-Sleep	-	0.9	-	mA
Deep-Sleep	-	10	-	μA
Power Off	-	0.5	-	μA

Tabla 2.5 Consumo Energético

2.2.5 Características de Radio WiFi

Parámetros	Mín	Típica	Máx	Unidad
Frecuencia de entrada	2412	-	2484	MHz
Impedancia de entrada	-	50	-	Ω
Reflexión de entrada	-	-	-10	dB
Potencia de salida de amplificador para 72.2 Mbps	15.5	16.5	17.5	dBm
Potencia de salida de amplificador para modo 11b	19.5	20.5	21.5	dBm
Sensibilidad	-	-	-	mA
DSSS, 1 Mbps	-	-98	-	dBm
CCK, 11 Mbps	-	-91	-	dBm
6 Mbps (1/2 BPSK)	-	-93	-	dBm
54 Mbps (3/4 64-QAM)	-	-75	-	dBm
HT20, MCS7 (65 Mbps, 72.2 Mbps)	-	-72	-	dBm
Rechazo a canal adyacente				
OFDM, 6 Mbps	-	37	-	dB
OFDM, 54 Mbps	-	21	-	dB
HT20, MCS0	-	37	-	dB
HT20, MCS7	-	20	-	dB

Tabla 2.6 Características de radio WiFi

2.3 Herramientas de desarrollo

2.3.1 Presentaciones comerciales

En el mercado se pueden conseguir los circuitos integrados ESP8266EX insertados en un módulo PCB, en el cual se agregan los elementos adicionales para su funcionamiento. Estos módulos son muy populares ya que son de muy bajo costo y poseen los GPIO accesibles para su uso directo. Además, el circuito integrado no viene en formato que no sea de soldadura superficial y por lo tanto no se justifica armar un módulo para su uso a baja escala.

Los módulos vienen en diversas presentaciones los cuales poseen diferentes factores de forma, separación entre pines (pitch) y cantidad de GPIO accesibles para conectar dispositivos. Si bien la empresa Espressif desarrolló un módulo para la venta, los más populares son los que comercializa la empresa Ali-Thinker. A continuación, en las siguientes tablas se pueden comparar las características de varios de los módulos más populares.

Nombre	Pines	Pitch	Factor de forma	Antena	¿Blindaje?	Tamaño (mm)
ESP-WROOM-02[11]	18	0.1"	2×9 DIL	trazo PCB	Si	18 × 20

Tabla 2.7 Modulo de Espressif

Nombre	Pines	Pitch	Factor de forma	Antena	¿Blindaje?	Tamaño (mm)
ESP-01	6	0.1"	2×4 DIL	trazo PCB	No	14.3 × 24.8
ESP-02	6	0.1"	2×4 almenado	conector U-FL	No	14.2 × 14.2
ESP-03	10	2 mm	2×7 almenado	Cerámica	No	17.3 × 12.1
ESP-04	10	2 mm	2×4 almenado	Ninguna	No	14.7 × 12.1
ESP-05	3	0.1"	1×5 SIL	U-FL conector	No	14.2 × 14.2
ESP-06	11	misc	4×3 dado	Ninguna	Si	14.2 × 14.7
ESP-07	14	2 mm	2×8 Agujero	cerámica + conector U-FL	Si	20.0 × 16.0
ESP-08	10	2 mm	2×7 almenado	Ninguna	Si	17.0 × 16.0
ESP-09	10	misc	4×3 dado	Ninguna	No	10.0 × 10.0
ESP-10	3	2 mm?	1×5 almenado	Ninguna	No	14.2 × 10.0
ESP-11	6	0.05"	1×8 agujero	Cerámica	No	17.3 × 12.1
ESP-12	14	2 mm	2×8 almenado	trazo PCB	Si	24.0 × 16.0
ESP-12E	20	2 mm	2×8 almenado	trazo PCB	Si	24.0 × 16.0
ESP-12F	20	2 mm	2×8 almenado	trazo PCB	Si	24.0 × 16.0
ESP-13	16	1.5 mm	2×9 almenado	trazo PCB	Si	W18.0 x L20.0
ESP-14	22	2 mm	2×8 almenado +6	trazo PCB	Si	24.3 x 16.2

Tabla 2.5 modelos de modulo ESP8266

Para este proyecto, por cuestiones de disponibilidad, se utilizó el módulo ESP-01, el cual posee 8 pines en total, 2 de ellos son de alimentación, VCC y GND, 2 corresponden a la interfaz de comunicación UART, Tx y Rx. Luego dispone del pin de *reset* y el de *chip select* (RST y CH_PD), los cuales son importantes para el funcionamiento del módulo y finalmente cuenta con dos pines de propósito general: GPIO0 y GPIO2.

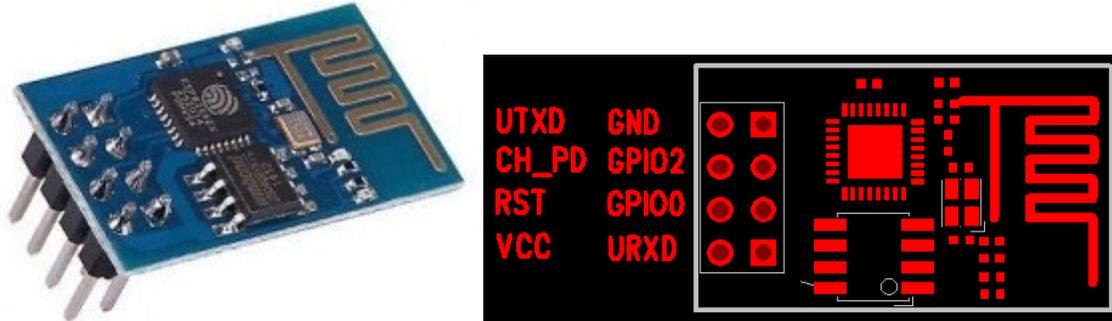


Figura 2.1 PCB ESP-01

2.3.2 Placa de desarrollo

El módulo ESP-01 posee dos modos de funcionamiento: uno es el modo normal, el cual el chip lee el programa grabado en la memoria flash. El otro es el modo de programación, el cual se usa para cambiar el firmware grabado en la memoria flash por otro que se transmite vía la interfaz UART.

Si bien GPIO0 y GPIO2 se pueden usar como entradas o salidas, al momento de iniciar el chip, su estado es importante para seleccionar los modos de funcionamiento del módulo.

GPIO0	GPIO2	Modo	Descripción
H	H	Flash	Inicia el Firmware almacenado en la memoria Flash
L	H	UART	Inicia en modo para cargar firmware vía UART

Tabla 2.6 Modos de inicio del ESP8266

Para garantizar el inicio en alguno estos modos, GPIO2 no debe estar conectado a masa, ya que al encender el módulo leerá un estado bajo y generará un error el cual no permitirá arrancar de ninguna forma usable.

Por otro lado, si GPIO0 ó GPIO2 son usados como entradas y al momento de iniciar están flotantes (no conectados), el módulo registrará el estado de los latch de lectura que posee el chip en cada entrada. Estos se encontraran en el nivel que leyeron al ser utilizados por última vez, lo cual generaría un inicio inestable. Para evitar esto, al usarse como entrada se debe procurar colocar el pin a masa o a VCC con una resistencia de *pull-up*, según corresponda, para iniciar correctamente.

Al usar GPIO0 y GPIO2 como salidas, no importa si se encuentran con una conexión flotante al iniciar, el chip leerá el inicio normal (estado alto ambos).

Teniendo en cuenta estas observaciones, se ha armado una placa de desarrollo que permite programar el módulo y probarlo de forma muy sencilla.

A continuación se muestra un diagrama esquemático de la placa de desarrollo. En este se conecta el módulo a una PC mediante un adaptador USB a TTL. Desde esta se pueden cargar firmwares, se pueden monitorear los puertos UART y enviar comandos para interactuar con el módulo durante su funcionamiento normal.

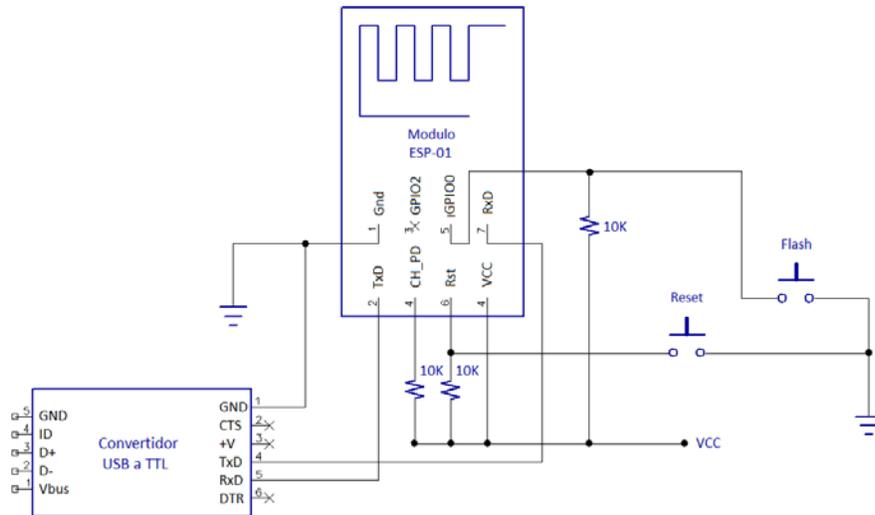


Figura 2.2 Circuito esquemático de placa de desarrollo ESP8266

Para energizar el módulo se prefiere una alimentación externa de 3,3V. Esto es debido a que al iniciar suele tener picos de corriente de aproximadamente 320mA (especialmente cuando está en modo softAP+Station) y si la fuente no puede suplir esta demanda el dispositivo no inicia.

El pulsador de reset permite reiniciar el dispositivo. Para acceder al modo flash, hay que mantener presionados ambos pulsadores, luego se suelta reset y por ultimo flash. De esta forma el dispositivo queda listo para la carga de firmware vía puerto UART.

Teniendo en cuenta que el ESP-01 tiene solo dos GPIO disponibles para configurar como entradas o salidas, es conveniente utilizar GPIO0 como entrada mientras se use el módulo en esta placa de desarrollo, ya que queda conectada al pulsador. Si fuera necesario usar GPIO0 como salida, el módulo debe retirarse de la placa de desarrollo antes de probarlo ya que se puede dañar ese pin de forma permanente.

2.3.3 Firmware

El firmware es el código que ejecuta el chip ESP8266EX. Como este microcontrolador no posee memoria para programa, el firmware se aloja en una memoria flash externa que se conecta mediante interfaz SPI. Esta memoria varía dependiendo el módulo utilizado, ya que la coloca el fabricante del mismo. En el caso del ESP-01 posee una memoria de 4Mbits.

Por defecto, los módulos vienen con firmware precargado que permite la operación de todas las funcionalidades de ESP8266EX mediante una lista de comandos AT. De esta forma el módulo depende de un microcontrolador que se conecte mediante la interfaz UART y pueda darle las instrucciones para el funcionamiento deseado. A su vez, el microcontrolador debe esperar la respuesta del ESP para continuar con la rutina. Por ejemplo, escanear las redes WiFi, conectarse a algún SSID, activar modo softAP, etc. El Firmware de comandos AT se ha ido actualizando con incluyendo cada vez mas comandos y nuevas funciones para el ESP8266.

Este modo de funcionamiento esclavo, dependiente de comandos AT, es útil para introducir comandos manualmente por un usuario a modo de explorar y familiarizarse con las capacidades del modulo. Pero a la hora de llevar a cabo un proyecto más complejo, nos encontramos con la dificultad de que debemos ocupar recursos del microcontrolador para enviar y recibir los comandos vía UART y se desperdicia el poder de procesamiento que posee el procesador Tensilica Xtensa LX106 del ESP8266.

En octubre de 2014, Espressif lanzo un SDK que permite programar el chip y que corra rutinas de forma independiente, quitando la necesidad de un microcontrolador maestro. Los SDK oficiales que mantiene actualizados Espressif son dos, uno basado en un sistema operativo de tiempo real (RTOS) y otro basado en *timers* y funciones *callback* (NON-RTOS), es decir, sin sistema operativo.

A partir de los SDK oficiales, se han desarrollado otros por terceros. Para este proyecto se consideró conveniente utilizar el SDK del ESP adaptado para funcionar en el IDE de Arduino. [18]

Es decir, que el código se escribe, se compila y sube al módulo ESP mediante el mismo software que se usa para hacerlo a una placa Arduino. Por lo tanto, se utiliza el mismo lenguaje de programación basado en C++ y también todas las funciones y librerías que se usan para proyectos en placas Arduino. [19]

2.4 Conexión a infraestructura WiFi

Una vez definidas las herramientas con las cuales se trabajará el módulo ESP ya se puede pasar al desarrollo del software para este proyecto. La principal función que llevara a cabo el módulo será la de establecer el acceso a la infraestructura WiFi. Para esto deberá:

- Escanear las redes disponibles
- Mostrar al usuario las redes disponibles
- Obtener del usuario la red y la contraseña para conectarse
- Establecer conexión a la infraestructura WiFi con los datos provistos por el usuario

En estas tareas se tratará de minimizar la intervención del usuario.

Este sistema está pensado para usar muchos dispositivos, por ende, integrarlos debe ser sencillo. Cada dispositivo posee un módulo de conexión WiFi y por lo tanto necesitan obtener el SSID y la contraseña de algún agente externo para conectarse a la red correcta. Si los dispositivos solo tomaran estos datos del usuario, como se suele usar en los equipos WiFi, cada vez que sea necesario conectarse a una nueva red WiFi la persona debería conectar todos los dispositivos que tenga.

Para eliminar esta dependencia y agilizar la instalación del sistema, se proponen dos opciones: la primera es usar el sistema de conexión WPS (*WiFi Protected Setup*), el cual permite asociar dispositivos a un AP mediante el pulsado de botones, lo cual resulta muy simple incluso para situaciones con varios dispositivos.

El proceso de WPS comienza al presionar el botón de *setup* en el AP. A partir de ese momento y por dos minutos el AP estará atento a algún dispositivo que quiera unirse a la red mediante WPS. Una vez que el AP está en este estado alerta, se debe pulsar el botón WPS en el dispositivo. Inmediatamente comienzan las negociaciones con el AP y momentos después el dispositivo queda conectado. Una vez que se conectó el AP abandona el estado de *setup* para volver al funcionamiento normal. Este proceso habría que repetirlo por cada equipo que se necesite asociar a la red. Este mecanismo de asociación no está disponible en todos los enrutadores por eso es necesario contar con alguna otra opción.

Como segunda alternativa se propone crear un mecanismo original para este proyecto, que siga un esquema donde se elija uno de todos los dispositivos como el encargado de interactuar con el usuario para obtener los datos de conexión. Una vez que este dispositivo se conecta correctamente a la red, compartirá el usuario y contraseña con los otros equipos que se quieran conectar. De esta forma el usuario solo interviene para conectar un dispositivo, el resto se conecta automáticamente.

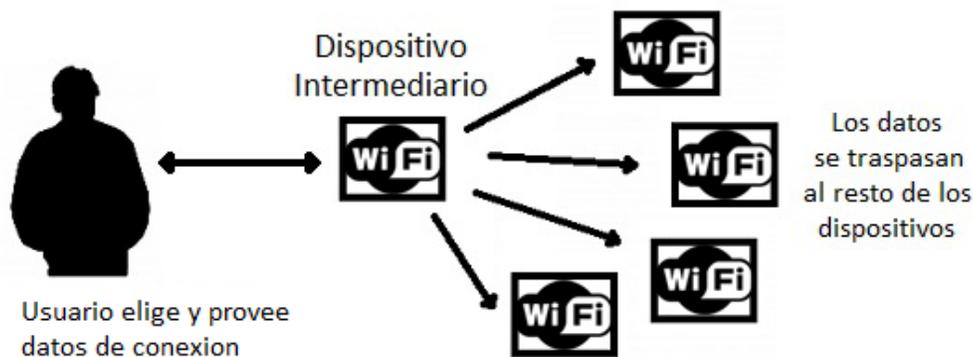


Figura 2.3 Esquema de asociación WiFi

En este esquema se representa como uno de los dispositivos es el encargado de interactuar con el usuario, mostrándole las redes WiFi, recibiendo la información de la red elegida y la contraseña. Luego este se encargará de difundirlo al dispositivo que se lo solicite. Los **dispositivos consultantes** son aquellos que se comunicarán con el intermediario para pedir los datos.

Más adelante se verá que el dispositivo que funciona como intermediario y difusor de los datos de conexión, también tendrá un papel especial una vez conectado a la infraestructura WiFi. Por eso, a partir de ahora, se llamara a este como el **dispositivo de referencia**. Debido al rol que toma en cuanto al acceso al medio y el direccionamiento en la red WiFi.

Para que este esquema funcione debe haber solo un dispositivo de referencia conectado a la infraestructura WiFi. Este, en principio podría cambiarse a voluntad del usuario o instalador mediante algún método de selección como puede ser un interruptor. Pero en la implementación que se llevó a cabo no se pudo concretar debido a la falta de más pines GPIO en el módulo ESP-01. Por lo tanto el rol de módulo de referencia no es intercambiable entre los dispositivos que componen este prototipo, pero está pensado para incluirse en futuras implementaciones.

2.4.1 Rutina de funcionamiento

A continuación se describirá paso por paso el proceso de asociación de un dispositivo a un AP WiFi. El equipo de **referencia** interactúa con el usuario para obtener la información y los **consultantes** le solicitan los datos de conexión. Cuando un dispositivo **consultante** solicita al de **referencia** el SSID (*Service Set Identifier*, nombre identificador de la red) y la PSK (*pre shared key*, contraseña de conexión), este último actúa como un difusor de los datos. Cuando los datos se obtienen por WPS, no hay ningún equipo intermediario ni difusor, por lo tanto todos operan con la rutina de **consultantes**. Sin embargo, en la etapa de direccionamiento seguirá siendo necesario un dispositivo de **referencia**, pero eso se desarrollará en el siguiente apartado.

Debido a que el modulo ESP-01 solo posee dos GPIO, se debió maximizar el uso de estas, por lo tanto tienen diferentes funciones a lo largo del proceso. Se usó GPIO0 como entrada conectada a un pulsador que tiene la función de "setup". GPIO2 se usó como salida conectándose a una luz LED, la cual durante este proceso de conexión indica en qué etapa se encuentra por medio de diferentes formas de parpadeo.

El proceso de asociación se podrá iniciar solo cuando los datos que se tengan almacenados no sirvan para conectarse a ninguna red WiFi. En esta situación el equipo estará en modo desconectado. Si el

dispositivo posee datos de WiFi válidos y se desea cambiarlos cuando la red está aun disponible, se deberá apagar dicho AP. Cuando el módulo se desconecta de la red WiFi vuelve a intentar conectarse automáticamente con los datos grabados, y si falla, permitirá activar el proceso de asociación a una nueva red.

En las Figuras 2.4 y 2.5 se presentan los diagramas de flujo de cada rutina y una explicación detallada de cada paso del proceso.

2.4.1.1 Rutina de conexión: Intermediario

A1: Al iniciar el dispositivo posee datos grabados en su memoria interna, incluso cuando es la primera vez que enciende. Automáticamente intentará conectarse usando esos datos.

A2: Si logra conectarse, significa que los datos que tiene grabados pertenecen a una red que está al alcance y son válidos. En ese caso se conectará y no dará lugar al proceso de *setup*.

Si no logra conectarse, puede ser porque la red no está al alcance y/o los datos son inválidos. Al detectar esta condición la luz LED comenzará a parpadear lento 3 veces.

A3: Durante el 3er parpadeo, la luz LED permanecerá encendida por más tiempo. Ese es el momento donde se detectará si el botón *setup* está presionado para dar lugar al proceso de configuración. Mientras parpadea y se espera la acción del usuario, el dispositivo sigue intentando conectar con los datos que tiene grabados, si llegara a conectar el proceso se interrumpirá y finalizará.

A4: Al pulsarse el botón de *setup* durante el momento adecuado se inicia el proceso de conexión mediante WPS. En este caso se recomienda haber ya presionado antes el botón WPS del enrutador/AP. Esto es debido a que, dependiendo del fabricante, no funciona si no es de esta forma. Hay que tener en cuenta que el periodo de atención en el cual el AP se encontrará receptivo de otro para la configuración de WPS es de 2 minutos.

A5: En el caso de tener éxito, el AP detectará la solicitud de conexión del dispositivo y comenzaran las negociaciones. Luego obtendrá los datos y los grabará en su memoria interna, sobrescribiendo los datos grabados previamente. Luego el proceso vuelve al estado A1.

Si no tiene éxito es porque el proceso WPS falló o porque el enrutador no es compatible con esta característica, entonces se necesitará la intervención del usuario. Si WPS se realiza pero falló, hay posibilidades de que los datos grabados hayan sido sobrescritos. Si WPS no se llevó a cabo, se conservarán los datos grabados con los que comenzó el proceso.

A6: En el caso de no concluir de forma positiva WPS, el dispositivo activará el modo softAP, en el cual funciona como enrutador habilitado por software. Luego de esto aparecerá en el área una nueva red WiFi que es la que se crea en esta instancia. El *SSID* y *PSK* de esta red son conocidos por el usuario o instalador.

También se inicia un servidor TCP en la dirección 222.200.50.250:8866, la cual es fija para todas las ocasiones. El usuario o instalador deberá usar un software de terceros para poder conectarse a este servidor. El programa debe operar como una consola TCP permitiendo conectar a un puerto determinado. Existen muchas opciones gratuitas como "Hercules" para Windows o "TCP/UDP Terminal" para Android.

El usuario se podrá dar cuenta que el dispositivo está esperando la conexión ya que encenderá de forma permanente la luz LED.

A7: Una vez creada la conexión por medio de softAP, el dispositivo queda a la espera que el usuario se conecte. Como en este proceso se requiere la intervención del usuario se supone que hay una persona supervisando estos pasos, no existe ningún tiempo de espera en esta instancia. El equipo queda en espera

indefinida a que el usuario se conecte a la red, por lo tanto, si se desea recomenzar hay que reiniciar el equipo presionando el pulsador *reset*.

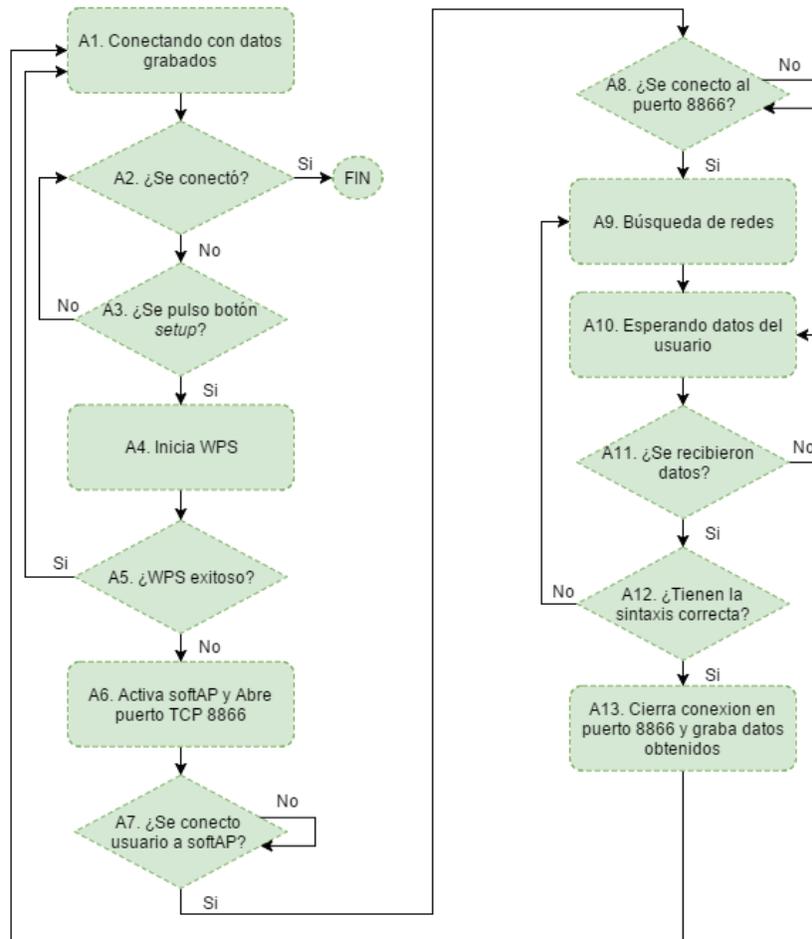


Figura 2.4 Diagrama de flujo: Dispositivo Intermediario

A8: Una vez detectada la conexión del usuario a la red, se espera que se conecte al puerto 222.200.50.250:8866. Por la misma razón que en el paso A7 no hay establecido un tiempo de espera y si se desea salir de esta instancia es necesario reiniciar el dispositivo por medio del pulsador *reset*.

A9: Al detectar la conexión de un cliente al puerto TCP 8866, el ESP8266 comienza inmediatamente con la búsqueda de redes WiFi disponibles en el área. Una vez recopilada esta información la envía al cliente conectado de forma que pueda visualizar los SSID y tenga la opción de elegir por alguno de ellos.

A10: En esta instancia, el dispositivo queda a la espera de que el usuario envíe los datos. Estos deben poseer una estructura determinada y contener ciertos señalizadores para que el ESP8266 pueda extraer los datos. Si se trata de una red oculta o simplemente quiere ingresar datos de una red que no está disponible, también lo podrá hacer. Esto siempre y cuando se respete la sintaxis adecuada.

El texto que se envía al puerto 8866 es procesado por un analizador sintáctico, o *“parser”*. Este software separa el texto partir de determinados patrones o *lexemas* que se encuentran en el texto fuente. [20] A partir de esta separación se pueden extraer diferentes tipos de datos de una misma cadena de caracteres.

Cada red escaneada tendrá un número asignado. Si el usuario quiere elegir una de esas redes deberá enviar al puerto 8866 el número anteponiendo los caracteres `{|NSN}` (*Network Selection Number*, Número de selección de red). O si el usuario prefiere enviar el nombre de la red lo podrá hacer anteponiendo `{|NID}` (*Network ID*, identificación de la red). Para enviar la clave se debe anteponer `{|PSK}` (*pre shared key*, clave pre compartida). El final del texto debe ser señalado con `{|}`. El mensaje completo quedaría de la siguiente forma:

`{|NID}NombreDeRed{|PSK}Contraseña{|}`
`{|NID}1{|PSK}Contraseña{|}` donde "1" sería el índice de la red elegida.

A11: En esta etapa no hay tiempo de espera, de modo que el usuario pueda revisar los datos e ingresarlos con tranquilidad. Al enviar algún texto al puerto TCP se pasará al siguiente paso de validación de sintaxis.

A12: En este paso se validará si la sintaxis es correcta. Concretamente se verificara que el texto enviado posea los patrones `{|NSN}` y `{|PSK}`, o `{|NID}` y `{|PSK}`. Si se llegaron a enviar los tres lexemas se tomará solo el dato de NID y se descartará el NSN. No es necesario que posea el finalizador `{|}` pero se recomienda para evitar errores de formato de texto.

Si el texto no valida estos requerimientos se vuelven a escanear las redes y se sigue el proceso desde el paso A9.

A13: Cuando la sintaxis del mensaje TCP es correcta, se extraen los datos del SSID y la contraseña. Luego se almacenan sobrescribiendo los datos anteriormente grabados. Se cierra la sesión con el cliente, pero el puerto se mantiene abierto al igual que el modo softAP, ya que se deja listo para que otros dispositivos se conecten y se les comparta estos datos recién obtenidos.

Antes de saltar al paso A1, el dispositivo apaga la luz LED que estuvo encendida durante el proceso de interacción con el usuario.

Luego de conectarse con los datos, la luz LED parpadeará cuatro veces indicando la conexión satisfactoria.

2.4.1.2 Rutina de conexión: Consultante

En la Figura 2.5 se muestra el diagrama de flujo de la rutina de obtención de datos de conexión de un dispositivo consultante, el cual recurrirá al dispositivo de referencia el cual le brindara los datos de conexión que recibió del usuario.

Los pasos **B1**, **B2**, **B3**, **B4** y **B5** inclusive son idénticos a los pasos **A1**, **A2**, **A3**, **A4**, **A5** de la rutina de conexión del dispositivo intermediario. Esto es debido a que esos pasos abarcan un camino común a la conexión por WPS. Como se explicó con anterioridad, si el AP cuenta con la capacidad de WPS y se decide conectar por este medio, no habrá necesidad de que ningún dispositivo interactúe con el usuario ni que posteriormente esté disponible para difundir la información de conectividad de WiFi. Esto es debido a que esa tarea queda relegada al propio AP mediante el proceso de WPS.

Se comenzará a describir paso por paso a partir de la etapa B6.

B6: Al fallar el proceso de WPS el dispositivo busca la red de softAP que proviene del equipo intermediario. Este nombre y clave para conectarse a esta red ya vienen pregrabados en el dispositivo consultante para este uso particular.

B7: Si no se conecta, significa que la red que pone en marcha el dispositivo de referencia esta fuera de alcance o incluso no hay ningún dispositivo configurado para esa función. En este caso no altera los datos grabados (paso B14) y regresa al paso B1 para intentarse conectar de nuevo.

B8: Si se conecta al softAP del dispositivo de referencia, tratara de conectarse como cliente al puerto TCP 8866.

B9: Si no se pudo conectar al puerto, pero si al softAP, posiblemente sea por un error del dispositivo de referencia, en este caso deja inalterados los datos de conexión (paso B14) y regresa al comienzo, a la etapa B1.

Si se conecta al puerto satisfactoriamente, encenderá la luz LED de forma permanente, indicando que está dialogando con dispositivo de referencia.

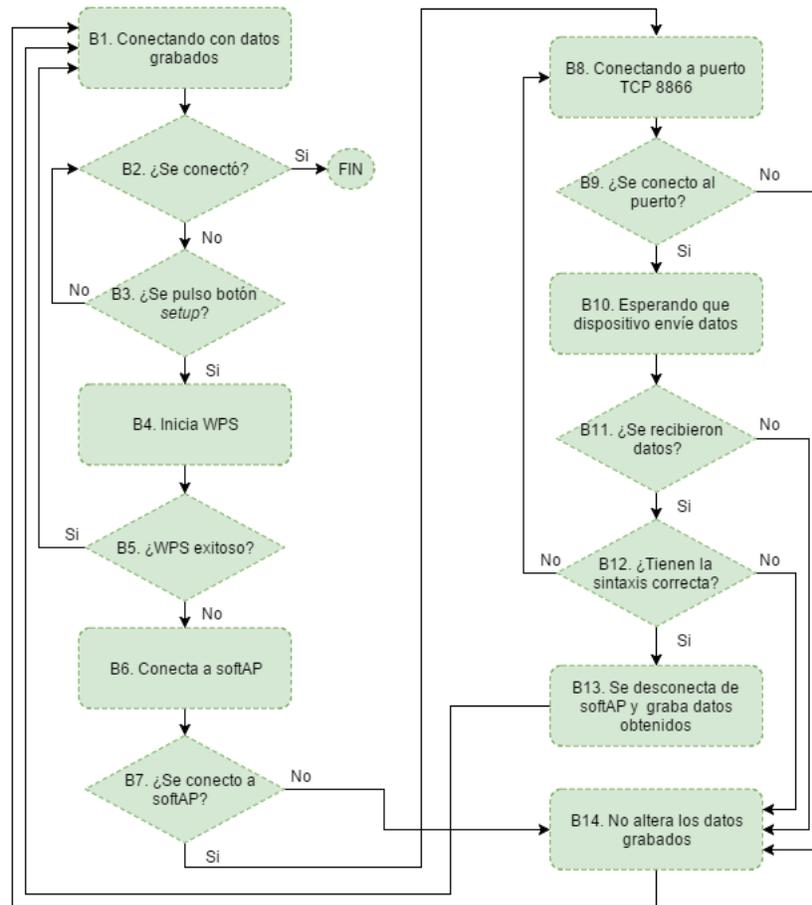


Figura 2.5 Diagrama de flujo: Dispositivo Consultante

B10: En este paso se espera que el dispositivo de referencia le envíe la información del AP elegido por el usuario para conectar. Para este caso habrá un tiempo de espera ya que si llegara a haber algún error, al no ser un proceso supervisado por el usuario, se desea que el dispositivo se recupere y regrese al diagrama de funcionamiento.

B11: Se considerará que el dispositivo de referencia no ha enviado el mensaje luego de un determinado tiempo de espera. En este caso preservan los datos de conexión previamente guardados (paso B14) y intentara conectarse de nuevo, recomenzando el proceso (paso B1).

Si reciben los datos, se revisará si la sintaxis es correcta. Se espera que el mensaje respete un formato determinado y que contenga alguno de los patrones mencionados en el paso A10 de la rutina de dispositivo de referencia.

B12: El mensaje que envía el dispositivo de referencia tendrá los lexemas { |NID}, el cual indica el nombre de la red y { |PSK} que indica la contraseña para conectarse. Se verificara que contenga estos dos señalizadores. De faltar alguno se considerará que los datos no tienen la sintaxis correcta. Se dejan inalterados los datos grabados de SSID y PSK y se saltea al paso B1.

Si la sintaxis es correcta se continúa al paso B13.

B13: Se graban los nuevos datos obtenidos reemplazando los antiguos, se desconecta del puerto 8866 y de softAP. De esta forma queda listo para conectarse con los nuevos datos obtenidos. Se apaga la luz LED indicando que el dialogo con el intermediario ha finalizado.

B14: Este paso no es una acción, sino una omisión. Se ha colocado para dar claridad al proceso. Es muy importante en caso de interrumpirse el flujo de configuración, por ser desatendido, que no se pierdan los datos pregrabados, los cuales pueden seguir siendo útiles en determinadas condiciones.

Si después del paso B1 se conecta satisfactoriamente a la red con los datos grabados, la luz LED parpadeará 4 veces indicando la conexión satisfactoria.

2.4.1.3 Rutina de difusión

El diagrama de flujo del proceso de difusión desde el punto de vista del dispositivo de referencia se presenta en la Figura 2.6. Este se comunica con el equipo consultante al solicitarle los datos de conexión.

Una vez que el dispositivo intermediario recibe los datos de conexión del usuario y se conecta satisfactoriamente a la infraestructura WiFi del AP, comienza a atender las tareas que tienen que ver con el sistema de domótica. Sin embargo, deja el modo softAP encendido y el puerto 8866 abierto. En el caso que algún dispositivo necesite los datos se conectara a ese puerto.

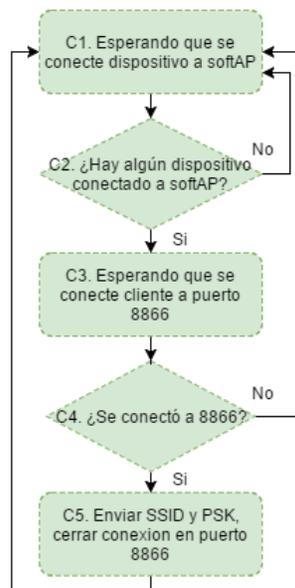


Figura 2.6 Diagrama de flujo dispositivo difusor

C1: Se puede considerar el estado de reposo del dispositivo de referencia en modo difusor. En este estado se encuentra conectado a la red anfitriona que proporciona el AP, atendiendo las tareas de la aplicación de domótica. Pero a su vez tiene *softAP* encendido, gracias a al funcionamiento dual que posee el ESP8266, en modo *station* y *softAP*. También tiene el puerto 8866 abierto a la espera de dispositivos que consulten por el SSID y PSK.

C2: El dispositivo de referencia nunca deja de monitorear si hay nuevas *stations* conectadas en su *softAP*. Esto significaría que un dispositivo consultante está en búsqueda de los datos de conectividad. Si no hay dispositivos conectados seguirá atendiendo el programa de domótica mientras que al final del ciclo verifica si hay nuevos equipos conectados a *softAP*.

Si se conecta un dispositivo al *softAP* continúa al siguiente paso.

C3: Si un dispositivo se conecta a *softAP* desatiende las tareas de domótica. Por esto se espera durante un tiempo a que el dispositivo se conecte al puerto 8866.

C4: Si pasa el tiempo de espera, atiende otro ciclo de las tareas de domótica regresando al estado C1. En caso de seguir conectado aguardará nuevamente durante el tiempo de espera establecido a que se conecte al puerto, y así sucesivamente.

Si se conecta al puerto dentro del tiempo estipulado pasa al siguiente paso.

C5: El módulo prepara el mensaje `{|NID}NombreDeRed{|PSK}Contraseña{|}` con los datos que actualmente está usando para conectarse a la infraestructura WiFi. Esta información se envía por el puerto 8866 y cierra la sesión con el dispositivo cliente volviendo al estado C1. Si el dispositivo consultante recibió los datos correctamente, en el próximo ciclo, se habrá desconectado del *softAP* para conectarse a la infraestructura WiFi que provee el enrutador.

Cuando este equipo se desconecta de la red WiFi se interrumpe el proceso de compartir los datos de conectividad hasta que se vuelva a conectar. En esa instancia el dispositivo pasa al estado A1 donde comienza el proceso de configuración. Ahí se brinda la posibilidad de cambiar los datos de conectividad si fuera necesario.

2.5 Direccionamiento y sistema de comandos

Una vez que los dispositivos están conectados a la red WiFi del enrutador, deben saber en qué dirección IP se encuentran los otros dispositivos con los cuales se necesitan comunicar. Para solucionar este problema se ha decidido precargar en los dispositivos consultantes una dirección IP fija la cual será la que usara el dispositivo de referencia una vez conectado a la infraestructura WiFi. Sabiendo la dirección IP de este dispositivo, los consultantes se podrán anunciar indicando datos importantes incluyendo la dirección IP que se les ha sido asignada actualmente. A partir de ahí, el dispositivo de referencia los registrará podrá comunicarse con ellos. Los siguientes pasos dependerán del programa de domótica, el cual se detallará en el siguiente Capítulo.

Si bien en cada red la dirección IP puede variar de un enrutador en otro, se sabe que los primeros 3 octetos de la dirección asignada por un mismo AP serán iguales para todos los dispositivos, y el último octeto es el que va variando de dispositivo en dispositivo.

El dispositivo de referencia debe ser el primero en conectarse a la infraestructura WiFi. Este cambiará la dirección IP que le fue asignada por el enrutador, manteniendo los 3 primeros octetos, pero el último lo cambiará a 250. De esta forma, si el AP le asigna una dirección 192.168.0.33, el programa en el ESP8266 automáticamente la cambiará a 192.168.0.250. Se eligió esta dirección 250 por ser un número elevado, normalmente un *pool* de DHCP no llega a las direcciones de más de 200 en el último octeto. De esta

forma se reducen las posibilidades de conflicto de dirección IP, pero no se eliminan. Es un requisito que no haya otro dispositivo en la red que use como dirección en el último octeto 250 para que este proceso funcione.

Cuando se conecta algún otro dispositivo a la red WiFi, sabrá que la dirección del dispositivo al cual debe reportarse es igual a la propia en los tres primeros octetos, pero el último es 250. Se usará del sistema de lexemas descrito en la sección 2.3.3.1.A10 incluyendo nuevos señaladores para confeccionar el mensaje de anuncio. Los patrones que indican diferentes tipos de datos o comandos serán profundizados en el siguiente Capítulo.

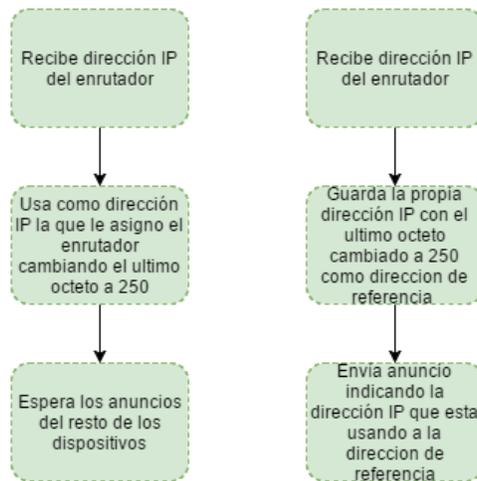


Figura 2.7 Rutina de direccionamiento.
Izquierda, dispositivo de referencia.
Derecha, dispositivo consultor.

2.5 Conclusiones

La completa funcionalidad WiFi y de pila TCP que provee el módulo ESP-01 con el chip ESP8266EX permitió delegarle toda la tarea de acceso a la infraestructura WiFi y conectividad TCP.

Se ha utilizado la capacidad de programación del chip para brindarle total autonomía y correr rutinas para asociarse a una infraestructura WiFi de cualquier enrutador mediante dos métodos diferentes WPS y uno propio de este proyecto.

Este método de conexión fue pensado para facilitar la tarea de conexión al usuario o instalador garantizando el mismo esfuerzo para uno o varios dispositivos. Utilizando un programa de conexión TCP de terceros, la persona se puede conectar a un dispositivo previamente elegido para la función de intermediario el cual se encontrará en modo *softAP*. Este modo consiste en funcionar como un AP, cuyo nombre y contraseña ya es sabido por el usuario y también está pregrabado en los otros dispositivos. Una vez conectado el usuario le puede enviar los datos de la red WiFi deseada para conectar mediante protocolo TCP.

Luego de pasarle los datos satisfactoriamente, además de usar esos datos para asociarse a la red WiFi, los guarda y comparte también mediante el modo *softAP*. De esta forma actúa como difusor de los datos de red. Por lo tanto el usuario en todo caso debe ingresar los datos una vez y el resto de los dispositivos obtienen los datos a partir del dispositivo de referencia.

Una vez conectados a la infraestructura del AP, los dispositivos que en su momento consultaron al dispositivo de referencia, se deben reportar a él pero a través de esta red y no del *softAP*. Así, le transfieren datos de interés y principalmente su dirección IP, para que los pueda registrar y enviarles datos por medio del protocolo TCP si el programa de domótica lo requiere. Esto es posible ya que los dispositivos consultantes conocen de antemano la dirección IP y el puerto al cual se tienen que anunciar.

Todos los mensajes, ya sean del usuario o del sistema, se envían por medio de conexiones TCP. Por esto, gracias a la programación de estas rutinas, el sistema domótico queda abstraído de lo que sucede en las capas de acceso al medio y las de transporte. Esto permite aprovechar los recursos del microcontrolador utilizado para la implementación de la interfaz de usuario o de algún otro dispositivo que lo requiera.

La dependencia que posee el sistema hacia el dispositivo de referencia le quita confiabilidad. Esto es porque sin el dispositivo de referencia no podrían conectarse a una nueva red o no podrían registrar su dirección para ser ubicados en el sistema de domótica. Sin embargo se ha decidido que es una medida necesaria en pos de simplificar el proceso de conexión. Para que el sistema no quede estrictamente centralizado se ha considerado la posibilidad de hacer que el rol de referencia sea intercambiable o rotativo entre los otros dispositivos que componen el sistema. Esto puede ser configurable por el usuario o instalador. De esta forma se le brinda más robustez al sistema.

CAPÍTULO 3

3.1 Introducción

En este Capítulo se presentará el desarrollo de la interfaz de usuario y los dispositivos periféricos. Estos aprovechan la plataforma de comunicación que provee el ESP8266, la cual fue descrita en el Capítulo anterior. El acceso al medio queda cubierto exclusivamente por los módulos WiFi, como así también la pila TCP. Por lo tanto esta parte se enfoca en la inteligencia de la administración de los dispositivos. Esto incluye como se lleva a cabo la comunicación al nivel de la aplicación de domótica y como se identifica cada equipo que participa en el sistema.

En las siguientes secciones se describirá la implementación de los métodos desarrollados en el Capítulo anterior y cómo se los integra al resto de las funciones que deben cumplir los dispositivos. Por eso se definirá tanto el hardware como software que se utilizará para lograr las características especificadas de cada equipo.

La interfaz de usuario se compone del módulo reconocedor de voz con el cual se identifican comandos que permiten navegar un menú donde se visualizan los dispositivos disponibles para operar. Esto último hace que se necesite de algún tipo de rutina para averiguar y registrar los equipos periféricos que están conectados al sistema. Luego se podrá comunicar con ellos mediante la plataforma que establece el módulo ESP8266.

En cuanto a los dispositivos periféricos, deben interpretar los mensajes que se le envíen para encender o apagar una luz LED. Esta señal de dos estados podría usarse para activar un relay que permitiría encender o apagar equipos de mayor potencia, pero esta etapa no se incluye en el modelo de prototipo que se desarrolla en este trabajo. Los periféricos también están preparados para recibir órdenes multiestado para las funciones que lo requieran, como puede ser operar un *dimmer* o leer el estado de un sensor.

3.2 Interfaz de usuario

La principal función de la interfaz es poner a disposición del usuario los diferentes periféricos que se encuentran conectados al sistema para poder interactuar con ellos mediante comandos de voz.

Estos se reconocen por un módulo específico para esa función, el EasyVR. A través de un display LCD se pueden visualizar los diferentes dispositivos que hay disponibles en el momento y el usuario puede navegar dicho menú mediante los comandos de voz.

El menú obtiene la información para mostrar de una memoria donde se encuentran grabados los datos de los periféricos asociados al sistema. La memoria se va completando por medio de la información que proviene de la comunicación con los dispositivos a través de la plataforma del módulo WiFi ESP8266.

A partir de las elecciones que va haciendo el usuario en el menú se disparan mensajes hacia los periféricos y luego se esperan respuestas de los mismos. La coordinación entre el menú, el módulo reconocedor de voz, la memoria y la interfaz de comunicación del ESP8266 queda a cargo de una placa de desarrollo Arduino Mega, la cual posee un microcontrolador ATmega2560 de Atmel. [21]

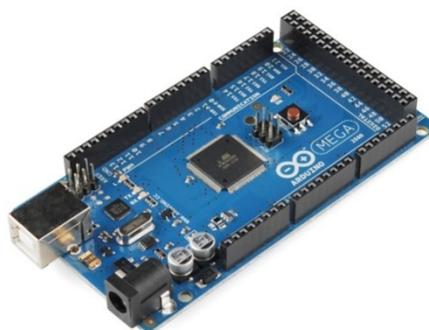


Figura 3.1 Placa de desarrollo Arduino Mega

La comunicación entre el módulo de reconocimiento de voz y el microcontrolador de la placa Arduino se hace por medio de una interfaz UART, al igual que con el módulo ESP8266. Para eso se necesitan dos puertos diferentes. Por esa razón se eligió este modelo de Arduino, ya que posee varios puertos de comunicación serie de forma nativa, es decir con hardware dedicado a esta función. El display se controla mediante una comunicación paralela a través de varios GPIO del microcontrolador ATmega2560. La memoria utilizada para guardar los datos de los dispositivos se encuentra integrada en el chip del ATmega2560.

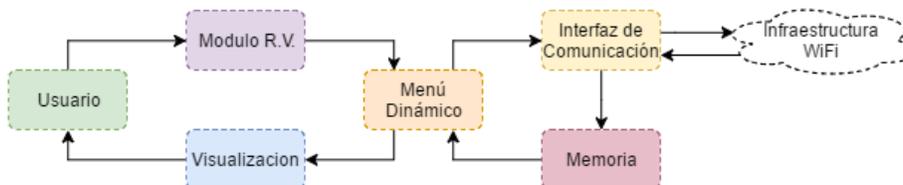


Figura 3.2 Diagrama en bloque de la interfaz de usuario: las flechas indican el sentido de la información

3.2.1 Gestión de dispositivos

Para que los periféricos que se van asociando a la red WiFi del enrutador puedan estar disponibles para utilizarse, es necesario que la interfaz de usuario los reconozca. De esta forma podrá mostrarlos en el menú y enviarle comandos para que actúen de acuerdo a las órdenes del usuario. Entonces debe haber algún mecanismo de descubrimiento que permita a la interfaz de usuario recolectar los datos de los periféricos. Estos datos son las características propias del equipo. Por ejemplo, si es un sensor o un actuador, si posee dos estados (como un switch o sensor de presencia) o múltiples estados (como puede ser un control o sensor de temperatura), etc. Uno de los datos más importantes que debe saber la interfaz de usuario es en qué dirección IP se encuentra el dispositivo para poder enviarle las órdenes.

Para averiguar qué dirección IP le fue asignada a cada periférico entra en juego la función del dispositivo de referencia que se introdujo en el Capítulo anterior. Como se explicó, este posee una dirección IP que es conocida por todos los otros dispositivos. Entonces estos le envían la dirección IP que se les fue asignada y el dispositivo de referencia va almacenando esos datos. En primera instancia la interfaz de usuario le debería solicitar a ese dispositivo de referencia una lista de todos los dispositivos que están conectados y que direcciones tiene cada uno. A partir de ahí podrá comunicarse directamente con ellos.

En este proyecto no se implementará al intercambiabilidad del dispositivo de referencia, debido a que, como se mencionó en el Capítulo 2, no hay una GPIO disponible para ese fin en cada dispositivo. En esta situación se consideró conveniente dar el papel de módulo de referencia al ESP8266 que se encuentra

en la interfaz de usuario. De esta forma la interfaz no tiene que consultar a otro dispositivo sobre las direcciones que están asignadas a los equipos “online”, sino que recibe directamente los anuncios de estos a medida que se van asociando al AP.

De nuevo se recalca que el esquema es abierto y no es estricto a esta implementación. En otras palabras, no necesariamente el dispositivo de referencia debe ser interfaz de usuario. Por ejemplo, puede haber varias interfaces de usuario y que ninguna de ellas sea el dispositivo de referencia, y se puede seleccionar un sensor para esa tarea, o un actuador, etc.

3.2.1.1 Flujo de información

Esta sección se enfocará en el envío y recepción de los mensajes en la interfaz de usuario. El origen de los datos recibidos que provienen de los dispositivos periféricos será cubierto en próximos apartados.

Hasta ahora se ha explicado que para este prototipo la interfaz de usuario se conecta a la red WiFi mediante un módulo ESP8266 que actúa como el dispositivo de referencia. Luego, a medida que los periféricos se van asociando a la red, le envían un mensaje de anuncio a la interfaz de usuario.

La gestión de la información que poseen estos mensajes se lleva a cabo en la capa de aplicación del modelo TCP/IP. Esto se fundamenta en el hecho que el texto que contiene la información se transmite encapsulado en paquetes TCP. El ESP8266 procesa los datos que provienen de la infraestructura WiFi y desencapsula la información que se transporta por protocolo TCP. Luego se la transmite al procesador de la interfaz de usuario por medio de comunicación serie UART. A partir de ese momento los datos quedan a disposición de la placa Arduino para gestionarlos, y el módulo ESP8266 sigue ocupándose de la rutina de comunicación.

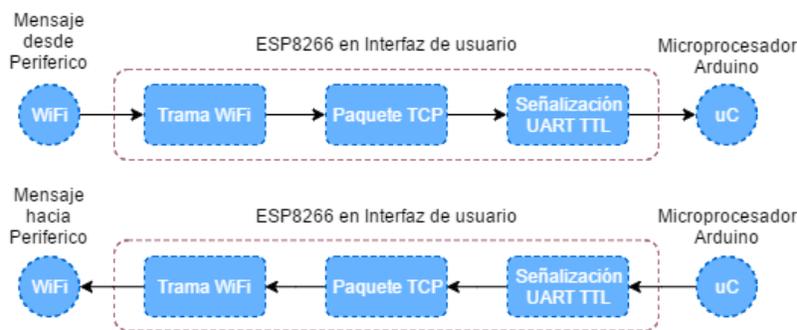


Figura 3.3 Diagrama de flujo del envío y recepción

Como el módulo ESP8266 trabaja de forma independiente de la placa Arduino, esta queda abstraída en el caso de algún problema con la red WiFi, por ejemplo si se desconectara o se saliera del rango de la red. En ese caso, la placa Arduino seguirá con su rutina habitual sin recibir mensajes ni respuestas del puerto serie. La rutina de reconexión va por cuenta del módulo ESP8266. Una vez restablecido el enlace WiFi también se restituirá la comunicación serie con la placa Arduino. Lo mismo sucede en caso de reiniciarse el módulo ESP8266.

Mensaje de anuncio y respuesta

En el Capítulo 2 se introdujo el uso de *lexemas*. Estos son patrones de caracteres que se incluyen en una cadena de caracteres mayor con el fin de señalar diferentes tipos de datos. De esta forma, mediante un software *parser*, se pueden extraer varios datos de interés en un solo envío. En aquel caso se utilizaron tres. Para el mensaje de anuncio que procesa la interfaz de usuario es necesario definir nuevos lexemas para poder incluir detalles sobre las características del dispositivo.

{|ANU|}: Este señalizador indica que la cadena de caracteres que se envía es un anuncio de un dispositivo periférico. Sirve para reconocer el motivo del mensaje, no contiene ninguna información para ser extraída.

{|MAC|}: Cada dispositivo debe poseer una identificación única que lo diferencie del resto que están asociados al sistema. Para esto se utiliza la dirección MAC de la interfaz WiFi que tiene cada equipo, ya que es única para cada uno. El procesador utiliza este dato para hacer referencia a cada dispositivo en su memoria interna. A continuación de este lexema viene la dirección MAC que será extraída por medio del parser.

{|NAM|}: Cada dispositivo debe poseer un nombre para que el usuario lo pueda identificar fácilmente entre los otros. El nombre será mostrado en el menú dinámico de la interfaz de usuario.

{|IPA|}: Este campo en el mensaje contiene la dirección IP que tiene asignada actualmente el periférico.

{|TYP|}: Este patrón indica el tipo de dispositivo que se está anunciando. Si es un actuador, sensor o interfaz de usuario. Si es de dos estados o multiestado, etc. De esta forma el dispositivo que recibe el anuncio sabrá que funciones son compatibles y podrá tratar con él de forma adecuada. Cada tipo se diferencia con un número. Para el caso de un interruptor el tipo es "1".

{ ANU }	{ MAC }	DirecciónMAC	{ NAM }	Nombre	{ IPA }	DirecciónIP	{ TYP }	NúmeroTipo	{ }
---------	---------	--------------	---------	--------	---------	-------------	---------	------------	-----

Figura 3.4 Composición final del mensaje de anuncio

El orden de los *lexemas* puede cambiarse, pero siempre el dato del campo debe estar a la derecha del que corresponda.

El dispositivo periférico debe saber si el mensaje que envió fue recibido correctamente para poder comenzar su rutina de funcionamiento habitual. Para eso es necesario que la interfaz de usuario le envíe un acuse de recibo.

{|AKN|}: Este lexema etiqueta el mensaje como un acuse de recibo.

{ AKN }	{ IPA }	DirecciónIPdestino	{ }
---------	---------	--------------------	-----

Figura 3.5 Composición del acuse de recibo

Este mensaje lo construye el procesador en la placa Arduino el cual lo envía por puerto serie al ESP8266 para que lo envíe por WiFi. Por eso debe incluir la dirección IP a donde va dirigido el acuse. El ESP8266 extrae esa dirección, remueve el campo de IP y lo envía al destinatario.

Comandos y respuestas

Los comandos son órdenes que envía la interfaz de usuario hacia los periféricos con el fin de alterar su estado actual y/o consultarlo. Pueden ser disparados a partir de una rutina programada o la interacción del usuario con el menú. Este prototipo solo utiliza comandos para modificar el estado de los dispositivos y son disparados a partir de la interacción con el usuario en el menú.

Esto significa que puede enviar dos tipos de mensaje, para encender o apagar el dispositivo. Una vez que se envían la interfaz de usuario esperara el acuse de recibo del dispositivo periférico. En caso de recibirlo en tiempo y forma considerará que el estado actual del dispositivo es el último que se envió.

{|SOH|}: Este lexema significa “colocar la salida en estado alto”, se utiliza para indicar a un periférico que posee un interruptor que lo encienda.

{|SOL|}: Al contrario que el anterior, significa “colocar la salida en estado bajo” e indica a un periférico que apague el interruptor.

Además de esta etiqueta el mensaje debe incluir también la dirección IP del dispositivo que se quiere accionar. Finalmente los mensajes quedan de la siguiente manera.

{ SOH }	{ IPA }	DirecciónIPdestino	{ }
{ SOL }	{ IPA }	DirecciónIPdestino	{ }

Figura 3.6 Comandos para encender y apagar interruptor

Una vez que se envían estos mensajes, la interfaz de usuario espera la confirmación de que el comando fue recibido y se procedió correctamente con la orden. Cuando lo recibe, actualiza en la memoria que el dispositivo se encuentra encendido o apagado, según el comando que se haya enviado. Esta respuesta contiene *lexemas* descritos con anterioridad.

{ AKN }	{ MAC }	DirecciónMAC	{ }
---------	---------	--------------	-----

Figura 3.7 Acuse de recibo que espera la interfaz de usuario

En este mensaje además del patrón de AKN, se espera la dirección MAC. Así se que se puede identificar de qué dispositivo periférico proviene la confirmación.

Lexema	Función	Datos Extraíbles
{ NID}	Nombre de red WiFi	SI
{ NSN}	Índice de red WiFi escaneada	SI
{ PSK}	Clave de acceso a la red WiFi	SI
{ MAC}	Dirección MAC del dispositivo	SI
{ IPA}	Dirección IP asignada	SI
{ NAM}	Nombre para visualizar	SI
{ TYP}	Tipo de dispositivo	SI
{ ANU}	Mensaje de anuncio	NO
{ AKN}	Acuse de recibo	NO
{ SOH}	Encender	NO
{ SOL}	Apagar	NO
{ }	Fin de mensaje	NO

Tabla 3.1 Resumen de *lexemas* utilizados en este proyecto

3.2.1.2 Memoria

A medida que la interfaz de usuario recibe mensajes de los periféricos va actualizando su memoria con diferentes datos que son importantes para la gestión de los mismos. Cuando se recibe el anuncio de un periférico que no se encuentra registrado se le asigna un bloque dos registros. Uno con los datos propios del periférico, y otro con los datos de control. El primero es el registro de las propiedades y el segundo es el registro de disponibilidad.

El registro de propiedades se actualiza mediante el envío y recepción de mensajes. Este se guarda en una memoria no volátil. De esta forma se conservan los datos a pesar de que se apague el dispositivo. Así la interfaz de usuario recordará los últimos dispositivos usados.

El registro de disponibilidad contiene los tiempos en cuales se envían los comandos y reciben los mensajes. Como estos datos son dependientes de un reloj interno del microcontrolador, el cual se reinicia cada vez que se corta la alimentación o se reinicia la placa, este registro se guarda en la memoria volátil del sistema. En la Figura 3.8 se muestra un esquema con los datos que contienen ambos registros y se explicará cada campo de los mismos.

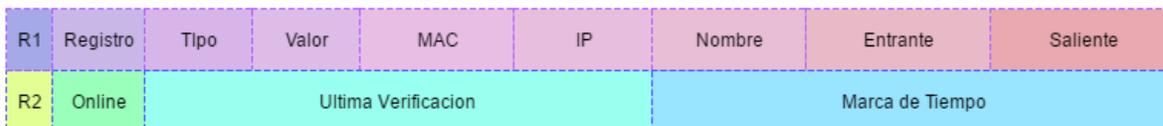


Figura 3.8 Bloque de registros de un periférico

Registro de propiedades

Cuando se recibe un mensaje de de anuncio o *ANU* primero se extrae la dirección MAC y se revisa si ya hay un bloque de registros asignado a ese equipo. Si ya hay, se actualizan los datos. Si no, se le asigna el primer bloque que este libre y se inicializan el registro de propiedades y el de control de disponibilidad. Si todos los bloques se encuentran utilizados, se grabarán en el primero cuyo dispositivo esta *offline*, sobrescribiendo los datos. Si todos están *online*, no se podrá registrar el nuevo dispositivo.

Luego de registrado el dispositivo, se esperará que mande periódicamente mensajes de anuncio para confirmar su estado de disponibilidad y a su vez actualizar los datos del registro de propiedades.

R1	Registro	Tipo	Valor	MAC	IP	Nombre	Entrante	Saliente
----	----------	------	-------	-----	----	--------	----------	----------

Figura 3.9 Registro de propiedades

Registro: Es una bandera que indica si este bloque de memoria que contiene ambos registros ha sido usado con algún equipo. Se encontrará en 1 si ya se utilizó o en 0 si no fue usado y por lo tanto está disponible para registrar un nuevo equipo.

Tipo: Aquí se almacena el valor numérico que contiene el mensaje ANU que identifica el tipo de dispositivo. Dependiendo del valor que contenga, la interfaz de usuario puede elegir con que rutina atender dicho equipo de manera de habilitar todas las funciones que este posea para su adecuado funcionamiento. En este proyecto este campo se registra pero no se lee, ya que todos los periféricos son tipo 1 que significa que son interruptores.

Valor: Aquí se coloca el valor actual que posee el periférico en su salida y/o entrada. En esta implementación, donde los periféricos son interruptores, este campo puede tomar dos valores: 0 si el interruptor está apagado o 1 si esta encendido. Este valor se actualiza luego de que se envía la orden de encender o apagar y se recibe el acuse de recibo de parte del periférico.

MAC: Contiene la dirección MAC del periférico, la cual se utiliza para identificarlo inequívocamente respecto a otros. Este valor se escribe cuando se inicializa el registro para un dispositivo determinado. La interfaz de usuario compara la MAC que se ha recibido en el mensaje de anuncio con los valores que ya se encuentran grabados en los registros. De esta forma, si coincide la MAC grabada en algún bloque de registros con la que se recibió significa que el dispositivo se encuentra registrado. Si no se encuentra la MAC recibida en ninguno de los registros utilizados, se graba en el primer registro que no se encuentre utilizado (Bandera de registro en 0).

IP: En este campo se guarda la dirección IP que actualmente tiene asignada el equipo. Este campo se graba cada vez que se recibe el mensaje de anuncio del dispositivo, de forma de actualizarlo si fue cambiado. Este campo se lee cada vez que la interfaz de usuario le quiera mandar un mensaje a un periférico determinado. Una vez obtenida esta información puede armar un mensaje para enviarlo.

Nombre: En este campo se guarda el nombre que tiene el periférico. Esta información se usa para que el usuario pueda identificar al periférico entre los otros. Normalmente se muestra en el menú el cual es dinámico, esto significa que se va actualizando constantemente. Por esa razón este campo se lee en cada ciclo de actualización del menú, de manera que si cambia se pueda mostrar inmediatamente.

Este campo se actualiza cada vez que la interfaz de usuario recibe un mensaje ANU. Para este proyecto, el nombre de los periféricos es fijo y viene pregrabado en cada uno. Sin embargo, en otras implementaciones este nombre podría ser personalizado por el usuario o incluso podría mostrar alguna información de interés de algún sensor. Impactando directamente sobre el menú.

Entrante: La finalidad de este campo es guardar el último mensaje que se recibió del dispositivo al cual pertenece el registro. Esto es para casos en los cuales no se pueda o no se desee procesar los datos en el momento que se reciben y se pueda posponer la tarea. Este campo no se usa en este proyecto, ya que la rutina que implementa la interfaz de usuario atiende los mensajes en el momento que se reciben mediante una interrupción. Este registro junto al saliente son los únicos que no se actualizan mediante el mensaje de ANU.

Saliente: Similar al campo anterior, pero en este caso se guarda el último mensaje que se envió. Esto puede ser útil para reenviar un mensaje en el caso de no haber sido recibido, o en el caso que se quiera implementar alguna rutina de automatización a lo largo del tiempo. En este proyecto se guarda en este campo el patrón `{|SOL}` o `{|SOH}` dependiendo cual haya sido la última orden enviada. Cuando se recibe el ACK del periférico se revisa el contenido de este registro y se actualiza el contenido del registro valor, según corresponda. Este registro junto al entrante son los únicos que no se actualizan mediante el mensaje de ANU.

En la Figura 3.10 se aprecian los diagramas de flujo que componen la rutina que administra el registro de propiedades de los periféricos. El mismo tiene tres ramas diferentes, ya que la intervención de este registro es disparada por tres eventos diferentes, recepción de un mensaje de anuncio, envío de un comando y recepción de un acuse de recibo.

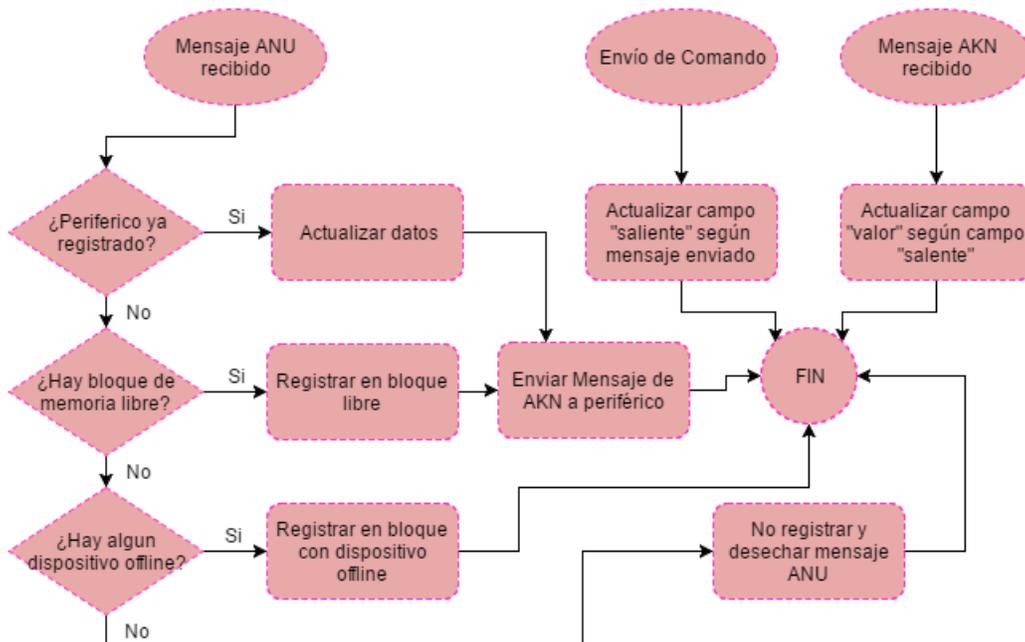


Figura 3.10 Diagrama de Flujo de registro de propiedades

Registro de disponibilidad

Una vez que la interfaz de usuario registra a los dispositivos periféricos disponibles es necesario llevar un control de su disponibilidad. De esta forma se puede saber cuales están listos para usarse y cuáles no. Luego en el menú se indica el estado de cada periférico para que el usuario lo tenga en conocimiento.

La disponibilidad de un equipo en el sistema significa que esté listo para recibir los comandos del usuario. Cuando se encuentre en este estado se dirá que se encuentra *online* o en línea. Para que se encuentre disponible es necesario que previamente se haya registrado, pero no es suficiente. El registro se lleva a cabo mediante el envío del mensaje ANU de parte del periférico hacia la interfaz de usuario, y luego que esta le envíe el acuse de recibo AKN. Si un dispositivo se encuentra registrado pero no está disponible se dirá que se encuentra *offline*, fuera de línea o desconectado.

Las razones por las cuales un dispositivo que ya fue registrado pueda estar offline son varias, pero las principales son que se encuentre apagado o fuera del alcance de la infraestructura WiFi. Por eso, mientras el dispositivo periférico se encuentre online debe enviar cada cierto tiempo un mensaje de ANU. Esto tiene dos funciones, la primera es que la interfaz de usuario pueda actualizar los datos del periférico, y la segunda es poder reiniciar un *timer* interno que es el que determina si el dispositivo sigue online o se desconectó.

La interfaz de usuario le asigna un *timer* a cada periférico que se registra. Este *timer* de verificación se coloca en cero luego de registrarse o de recibir un mensaje de ANU del periférico. Si este no se anuncia nuevamente antes que el tiempo del *timer* llegue a un determinado valor, la interfaz de usuario considera que ese dispositivo esta offline.

Por otro lado, cuando se le envía un comando a un periférico y este no responde con el acuse de recibo AKN, también se reconoce que ese dispositivo está desconectado. El mensaje de AKN se aguarda durante un tiempo que controla otro timer que es el de tiempo de espera. Cada periférico tiene un timer de tiempo de espera asignado diferente.

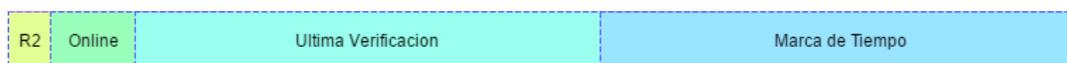


Figura 3.11 Registro de disponibilidad

Online: Es una bandera que indica si el dispositivo esta online. Si se encuentra en 1, el dispositivo esta *online*, si se encuentra en 0 esta *offline*. Esta bandera se mantiene actualizada mediante los diferentes mecanismos de control de disponibilidad que se explicaron anteriormente.

Luego de estar un tiempo apagada, la interfaz de usuario no tiene información confiable si los periféricos se encuentran *online*, pero recuerda los que fueron registrados. Esto es porque el registro de propiedades se guarda en una memoria no volátil. En este caso se supone que todos los dispositivos que se encuentren registrados estarán online en el momento de encender la unidad. Esto significa que cuando se inicializa la interfaz de usuario se verifica cuales registros están utilizados (con la bandera de registro en 1) y entonces también se coloca la bandera online en 1.

Ultima Verificación: El microcontrolador posee un reloj interno el cual va aumentando su valor a medida que transcurre el tiempo de funcionamiento de la placa Arduino. En este campo se guarda el valor que tiene ese reloj la última vez que se verifico que el periférico estaba funcionando. Esto es cuando se recibe un mensaje de ANU o un mensaje de AKN.

En cada ciclo de verificación se compara el valor de este registro con el tiempo actual que marca el reloj interno de la placa Arduino. Si se pasó el tiempo preestablecido para el cual se espera una nueva señal del periférico, se cambia la bandera *online* al valor 0.

Marca de Tiempo: En este campo se guarda el valor del reloj en el momento que se envió un comando. Este valor se compara con el tiempo actual del reloj interno del Arduino y a partir de esa comparación se determina si se pasó el tiempo de espera para recibir el acuse de recibo de parte del periférico.

En la Figura 3.12 se presenta el diagrama de flujo del proceso de control de disponibilidad. El mismo inicia desde el bloque del estado *online* u *offline*, según sea el estado inicial del registro. Luego finaliza también en uno de esos dos bloques. Si este diagrama se incluyera como parte de uno mayor, luego de llegar a uno de los dos estados se continuará con el flujo hacia los otros procesos. Pero desde el punto de vista del proceso de control de disponibilidad es cíclico, una vez que llega a *online* u *offline*, verifica de nuevo la situación y así sucesivamente.

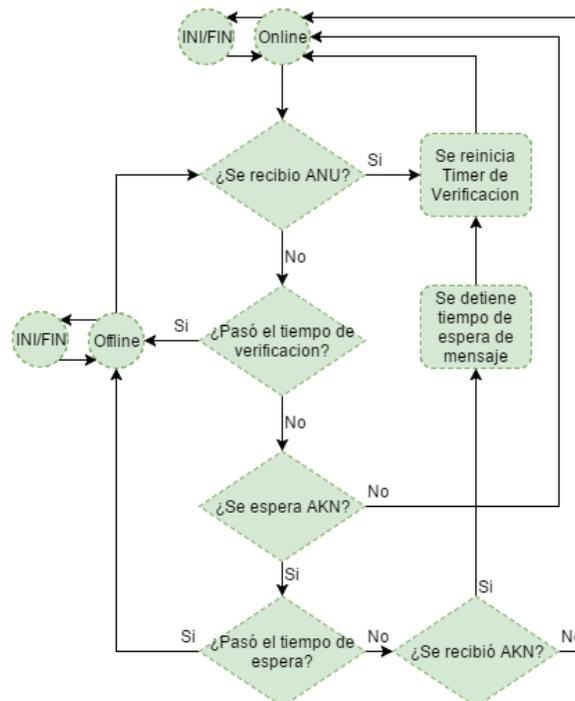


Figura 3.12 Diagrama de Flujo del control de disponibilidad

3.2.2 Menú dinámico

La interfaz de usuario posee un display LCD por el cual muestra la información del sistema y permite al usuario ver los periféricos disponibles para interactuar con ellos. El menú es el formato con que se presenta el conjunto de información y las opciones a elegir. La manera en la cual la persona navegará el menú es mediante comandos de voz.

Para hacer esta tarea lo más sencilla posible se han indexado las opciones que muestra el menú. De esta forma, cuando el usuario quiera usar una determinada opción, dirá el número del índice y entonces el sistema procederá con el procesamiento de la orden.

La estructura del menú varía de acuerdo a la página en el cual se encuentra, y el contenido que muestra proviene de los bloques de registro de los dispositivos, por lo tanto depende del estado de estos.

En la página raíz del menú se muestran un total de ocho opciones las cuales representan la cantidad total de periféricos que puede administrar la interfaz de usuario sin tener que paginar el menú principal (solo limitado por el tamaño de la pantalla). El elemento del menú que represente un dispositivo registrado mostrara su nombre, mientras que los que aún no tengan un periférico asignado figurarán como "NO REG."

Una vez que el usuario accede a algún elemento, se pasa a una página del menú donde se muestra si el dispositivo está *online* u *offline*, si se encuentra encendido u apagado y se muestran las opciones de encender, apagar o volver a la página raíz. En la Figura 3.13 se aprecian fotografías del menú navegable.

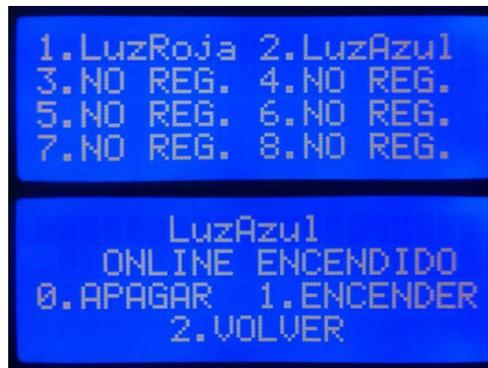


Figura 3.13 Funciones del Menú: Arriba: página raíz. Abajo: página de periférico

Los detalles de cada periférico y el nombre de cada uno que aparecen en la página raíz provienen de la información guardada en los bloques de registros. Este tipo de contenido que se muestra el menú es independiente del proceso que administra el cambio de páginas. La actualización de la información es constante en cada ciclo del proceso principal. Por lo tanto, a pesar de dejar el menú estático en una página, el contenido se irá actualizando casi de forma instantánea a medida que se actualiza la información en los registros de propiedades y de disponibilidad. En la Figura 3.14 se presentan los diagramas de flujo de ambos procesos, el de actualización de contenido y el de navegación del menú.

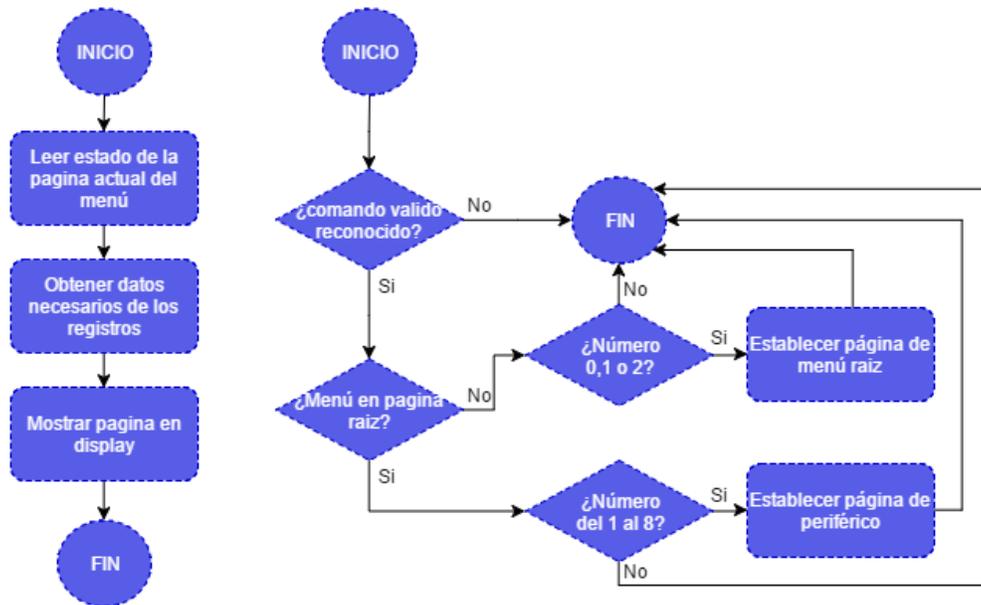


Figura 3.14 Izquierda: Diagrama de Flujos de actualización de contenido
 Derecha: Diagrama de Flujos de navegación de menú

3.2.3 Módulo reconocedor de voz

El medio que se eligió para que el usuario pueda navegar el menú y controlar los periféricos es el de reconocimiento de voz. Esto es porque se consideró la opción más versátil que cumple con el objetivo de minimizar el esfuerzo físico para operar el sistema.

Para llevar esto a cabo, la unidad de interfaz de usuario cuenta con el módulo de reconocimiento de voz, EasyVR de Veeear. El mismo posee gran cantidad de características. La más importante es la de reconocimiento de determinados comandos de voz que pueden provenir de cualquier locutor. Esto sin precisar de un entrenamiento previo ni conexión a internet y es compatible con el idioma español. En la Tabla 3.2 se aprecia el listado de los comandos independientes del locutor que es capaz de identificar el módulo EasyVR.

Trigger Word set	Command Index	Language					
		0	1	2	3	4	5
		English (US)	Italian	Japanese (Rōmaji)	German	Spanish	French
0	0	robot	robot	ロボット <i>robotto</i>	roboter	robot	robot
	1	action	azione	アクション <i>acution</i>	aktion	acción	action
	2	move	vai	進め <i>susu-me</i>	gehe	muévete	bouge
	3	turn	gira	曲がれ <i>magare</i>	wende	gira	tourne
	4	run	corri	走れ <i>hashire</i>	lauf	corre	cours
	5	look	guarda	見ろ <i>miro</i>	schau	mira	regarde
	6	attack	attacca	攻撃 <i>kougeki</i>	attacke	ataca	attaque
	7	stop	fermo	止まれ <i>tomare</i>	halt	para	arrête
1	0	hello	ciao	こんにちは <i>konnichiwa</i>	hallo	hola	salut
	1	left	a sinistra	左 <i>hidari</i>	nach links	a la izquierda	à gauche
	2	right	a destra	右 <i>migi</i>	nach rechts	a la derecha	à droite
	3	up	in alto	上 <i>ue</i>	hinauf	arriba	vers le haut
	4	down	in basso	下 <i>shita</i>	hinunter	abajo	vers le bas
	5	forward	avanti	前 <i>mae</i>	vorwärts	adelante	en avant
2	0	backward	indietro	後ろ <i>ushiro</i>	rückwärts	atrás	en arrière
	0	zero	zero	ゼロ <i>zero</i>	null	cero	zéro
	1	one	uno	一 <i>ichi</i>	eins	uno	un
	2	two	due	二 <i>ni</i>	zwei	dos	deux
	3	three	tre	三 <i>san</i>	drei	tres	trois
	4	four	quattro	四 <i>yon</i>	vier	cuatro	quatre
3	5	five	cinque	五 <i>go</i>	fünf	cinco	cinq
	6	six	sei	六 <i>roku</i>	sechs	seis	six
	7	seven	sette	七 <i>nana</i>	sieben	siete	sept
	8	eight	otto	八 <i>hachi</i>	acht	ocho	huit
	9	nine	novе	九 <i>kyu</i>	neun	nueve	neuf
	10	ten	dieci	十 <i>jyuu</i>	zehn	diez	dix

Tabla 3.2 Palabras que se reconocen de cualquier locutor

Además, se pueden usar comandos personalizados por el usuario, pero en este caso son dependientes del locutor y requieren de un entrenamiento previo del módulo. En este proyecto, no se hará uso de estos comandos por cuestiones de practicidad y robustez. Sin embargo, esta capacidad puede ser usada en futuras mejoras para que el usuario pueda personalizar el uso del dispositivo.

El módulo viene preparado para adaptarse perfectamente a la plataforma Arduino. El fabricante pone a disposición librerías para usar en el IDE de Arduino y también provee un *shield* el cual permite conectar directamente el módulo a la placa de desarrollo tomando la tensión de alimentación que necesita de esta sin la necesidad de conectores o placas adicionales.

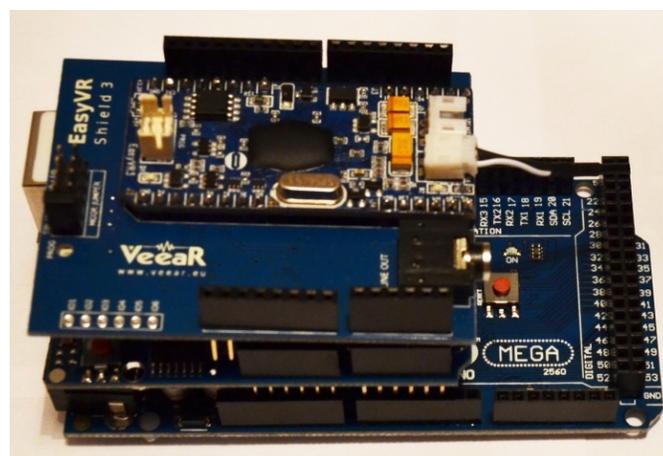


Figura 3.15 Shield EasyVR encastrado en Arduino Mega

El módulo EasyVR se comunica con la placa Arduino a través de un puerto serie y trabaja en modo esclavo. Mediante esta comunicación el microcontrolador transmite comandos hacia el modulo. Estos disparan las rutinas de reconocimiento para grupos de comandos específicos. Estos grupos, cuando contienen comandos independientes del usuario, son llamados *wordset*. En la tabla 3.2 se presentó la lista de estos comandos y los diferentes *wordsets* a los cuales pertenecen. Por ejemplo, el *wordset* 2 contiene direcciones, arriba, abajo, izquierda, derecha, adelante y atrás. Cuando se indica que reconozca un comando del *wordset* 2, el módulo escucha durante un tiempo de espera configurable a que el usuario diga una de esas palabras. Si no se detecta devolverá un código de error y de lo contrario le transmitirá por puerto serie un número que representa la palabra identificada. Mientras el módulo se encuentra escuchando al usuario, el microcontrolador espera que termine el proceso de identificación. Este tiempo es variable, dependiendo de en qué momento se detecte la palabra. A lo sumo, si no se detecta ninguna palabra, tardará el tiempo de espera que se haya establecido. En todo caso, mientras dure el proceso, el microcontrolador queda retenido. Luego, el resultado de la detección queda guardado en el módulo EasyVR. Antes de continuar el microcontrolador debe solicitarle al modulo el resultado de el reconocimiento, de lo contrario esa información se puede perder al disparar otra rutina de reconocimiento.

Se buscó que el dispositivo esté siempre alerta a las palabras del usuario para que pueda reaccionar de inmediato a los comandos. Sin embargo, como los recursos del microcontrolador quedan ocupados al 100% mientras el módulo EasyVR escucha, no se puede tomar un tiempo de espera infinito, ya que el microcontrolador debe atender los otros procesos del sistema. Para ello se ha empleado un *timeout* de 10 segundos, lo cual es suficiente para que registre los comandos de voz. Luego de este intervalo de escucha, el microcontrolador atenderá los otros procesos, como lo son la gestión de periféricos, la visualización del menú y la comunicación. Esto se encuentra en un bucle permanente, por lo tanto al finalizar la escucha atiende los procesos y luego vuelve a escuchar comenzando de nuevo el ciclo. Este bucle es la rutina principal que realiza la interfaz de usuario de la cual se derivan el resto de los procesos.

El módulo posee un sistema de adaptación al ruido ambiente, lo cual le permite funcionar tanto en ambientes sin ruidos, con una voz sola con varias voces superpuestas o incluso en ambientes ruidosos. Sin embargo, se ha encontrado el inconveniente que si el ruido ambiente o la cantidad de voces fluctúa abruptamente, hasta que el módulo se adapta a las nuevas condiciones de ruido puede detectar falsos positivos. También puede suceder, que en el ambiente se pronuncie alguna de las palabras que el módulo espera reconocer. Esto puede pasar durante una conversación con otra persona, o mediante una voz en la TV, etc. En ese caso se produciría un falso positivo también, ya que si bien la palabra fue detectada correctamente porque se pronunció, no fue con intenciones de comandar el menú.

Los falsos positivos conducen a una navegación espontanea del menú. Para evitarlos, se implementó un bloqueo del menú. Mientras está bloqueado, no se detectan palabras. Para desbloquearlo se debe pronunciar un sonido o palabra especial designada para tal fin. Una vez que se usó algún periférico, este se vuelve a bloquear automáticamente. En el prototipo se implementó el desbloqueo mediante el sonido de dos palmas o dos chasquidos de dedos. Si otra implementación lo precisara, se podría cambiar a un silbido, o mediante el uso de alguna palabra personalizada previamente entrenada como un comando dependiente del locutor.

El menú muestra las opciones indexadas, una vez que se encuentre desbloqueado se espera que el usuario diga una palabra del *wordset* 3. Este corresponde a los números del 0 al 9. El número que pronuncie el usuario se interpreta como el índice de la opción que desea acceder. El módulo procesa la voz y transfiere el resultado. El microcontrolador recibe la detección procede según el estado actual del menú.

Habiéndose descrito la rutina principal de la interfaz de usuario, en la Figura 3.16 se aprecia el diagrama de flujo el cual abarca la inicialización y el bucle de atención de los procesos.

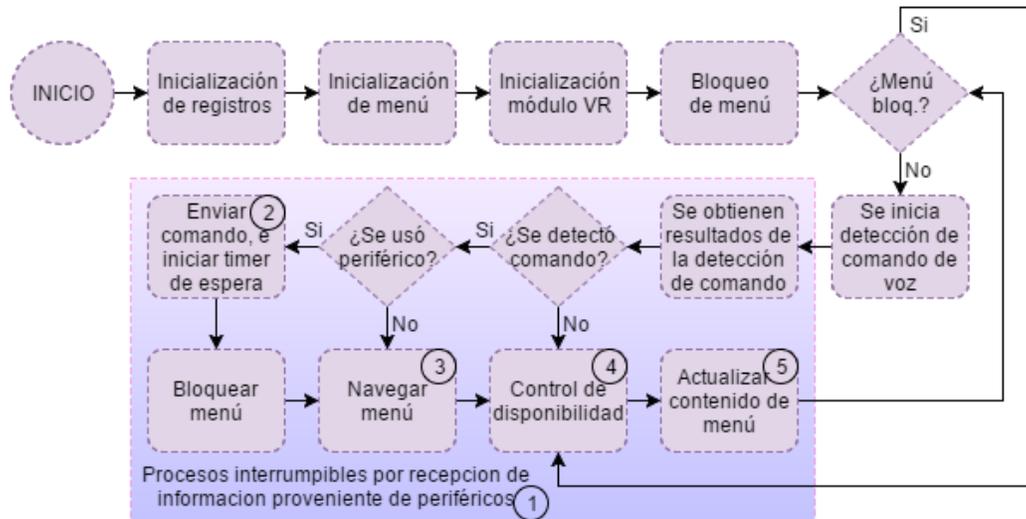


Figura 3.16 Diagrama de Flujo principal de la interfaz de usuario

- 1) Incluye diagrama de flujo de recepción (apartado 3.2.1.1), y en caso de ser un mensaje AKN sigue con el de registro de propiedades (apartado 3.2.1.2).
- 2) Diagrama de flujo de envío (apartado 3.2.1.1) y diagrama de flujo de registro de propiedades (apartado 3.2.1.2).
- 3) Incluye diagrama de flujo de navegación de menú (apartado 3.2.2).
- 4) Incluye diagrama de flujo de control de disponibilidad (apartado 3.2.1.2).
- 5) Incluye diagrama de flujo de actualización de contenido (apartado 3.2.2)

Este es el diagrama de flujo raíz o principal, del cual se desprenden los otros diagramas de flujo que se han incluido en anteriores secciones. Cuando se recibe un mensaje de un periférico por la interfaz WiFi del ESP8266, este desencapsula los datos del paquete TCP y lo transmite por comunicación serie a la placa Arduino. Esto genera una interrupción la cual se atenderá inmediatamente si el flujo se encuentra dentro del área sombreada, de lo contrario se atenderá cuando el flujo reingrese en dicha área. La rutina que atiende esta interrupción es la del flujograma de *registro de propiedades* (apartado 3.2.1.2), el cual actualiza los datos de los bloques de registros en la memoria.

De la misma forma los bloques de *control de disponibilidad*, *estado de menú*, y *contenido de menú* se expanden a los respectivos diagramas homónimos presentados en apartados anteriores.

Consideraciones de uso

Se ha comprobado empíricamente durante el desarrollo del proyecto que el módulo de reconocimiento de voz es susceptible a interferencias que son generadas por transmisores de radiofrecuencia al encontrarse próximos al equipo. Estas interferencias alteran el funcionamiento del reconocimiento de voz, generando constantes errores en la detección a pesar de estar en las mejores condiciones. Esto es sin ruido ambiente y con voz clara y fuerte.

Se ha logrado mitigar el efecto colocando un blindaje al módulo WiFi ESP8266 en la unidad de reconocimiento de voz, sin embargo no se ha podido eliminar por completo. Para que el funcionamiento sea normal, debe tenerse en cuenta que la interfaz de usuario debe estar por lo menos a 50cm de distancia de cualquier transmisor de radiofrecuencia, como por ejemplo, otro dispositivo WiFi o Bluetooth.

El inconveniente no genera falsos positivos y en condiciones normales de funcionamiento no altera su efectividad. Por esto se ha decidido implementar el equipo en estas condiciones y postergar la solución para futuras revisiones del sistema.

3.2.5 Hardware de interfaz ESP8266/Arduino

El microcontrolador de la placa Arduino Mega requiere alimentación de 5V y el módulo ESP8266 usa 3,3V. Si bien la placa Arduino posee salida de 3,3V esta no satisface los requisitos de corriente de alimentación que necesita el ESP8266 ya que solo puede dar una corriente máxima de 50mA y el módulo puede tener picos de consumo durante el funcionamiento normal de 170mA y durante el inicio de hasta 320mA.

Para esto fue necesario colocar un regulador de tensión para alimentar al módulo ESP8266 que supla los requerimientos de corriente que necesita. La entrada del regulador se conecta a la alimentación de la placa Arduino que es de 5V y la salida, que provee los 3,3V, se conecta a la alimentación del módulo ESP8266.

También surgen inconvenientes en la comunicación entre el módulo WiFi y la placa Arduino debido a los niveles lógicos que usa cada circuito. Luego que el módulo ESP8266 desencapsula el mensaje del paquete TCP se lo transmite al puerto de comunicación del Arduino. Esta comunicación se realiza mediante un puerto UART.

Arduino maneja una lógica de 5V mientras que el ESP8266 utiliza lógica de 3,3V. Como se ve en la Figura 3.17, en principio estas dos familias lógicas son compatibles. [22] Sin embargo se han registrado problemas cuando el ESP8266 envía datos al puerto del Arduino. Estos problemas surgen por no poder mantener niveles adecuados para que el receptor pueda leer en un 1 lógico.

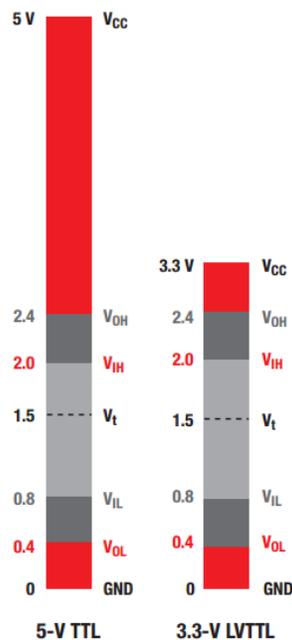


Figura 3.17 Niveles lógicos asociados a cada tensión

3.3 Dispositivos periféricos

Los dispositivos periféricos son los equipos que intervendrán para llevar a cabo las acciones en concreto que el usuario quiera realizar con la asistencia del sistema. Muchas veces estas tareas no son solo a un nivel eléctrico, sino que pueden ser mecánicas y/o físico-químicas. Por ejemplo, para el levantamiento de una persiana, además de la electrónica para el procesamiento de la orden, es necesario un agente que interactúe mecánicamente con la persiana para generar el movimiento adecuado, como un motor. Este a su vez necesita de la electrónica apropiada para ser controlado. También, si se desea conocer la temperatura de una habitación, se necesita de un sensor que transforme la variable de interés a una variable eléctrica. Luego una unidad de procesamiento tendrá que traducir la señal a un formato que se pueda enviar a la interfaz de usuario.

Para la implementación de este prototipo no se ha incluido en los periféricos las partes del sistema necesarias para interactuar con algún elemento o variable que se pueda encontrar en el hogar. Esto escapa el alcance del proyecto, pudiéndose transformar motivo de otros desarrollos. Sin embargo, para probar el funcionamiento del sistema se han construido dos periféricos actuadores de dos estados. El estado de la salida de cada actuador se puede ver mediante una luz LED que se enciende o apaga.

Las características de los dispositivos periféricos son las siguientes:

- Funcionar como dispositivos consultante (Capítulo 2). Esto incluye:
 - Obtener nombre y contraseña de WiFi del dispositivo de referencia.
 - Asociarse a la red WiFi una vez obtenidos los datos.
 - Enviar mensaje de anuncio al dispositivo de referencia.
- Procesar órdenes de la interfaz de usuario para encender o apagar la luz LED.
- Una vez procesada la orden, enviar un acuse de recibo a la interfaz de usuario para informar el éxito de la operación.
- Enviar periódicamente mensajes de anuncio a la unidad de interfaz de usuario para que mantenga registro de su disponibilidad.

3.3.1 Componentes del Hardware

Para llevar a cabo las funciones requeridas se aprovechan las características del módulo ESP8266 que le permiten ejecutar rutinas programables. Como se necesita solo una GPIO configurada como salida para construir un actuador tipo interruptor, se puede usar la GPIO2 del ESP8266 para tal fin. De esta forma basta con un módulo ESP8266 para realizar todas las funciones del periférico. Esto quiere decir que el módulo no solo se ocupa de la conexión WiFi, sino que también de las funciones de actuador del sistema domótico.

La alimentación del circuito se realiza por medio de dos pilas AA que satisfacen los requerimientos de corriente del equipo. Además, de esta forma no es necesario conectar a la corriente alterna de línea, pudiendo aprovechar al máximo la portabilidad del periférico.

La Figura 3.19 presenta el circuito esquemático de un equipo periférico.

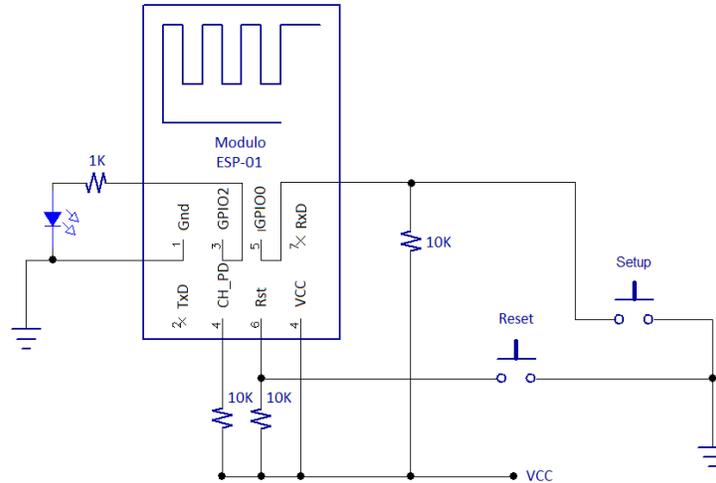


Figura 3.19 Esquemático del circuito de un periférico

En este caso, los puertos de transmisión y recepción no se utilizarán ya que no se requiere ningún tipo de comunicación con otro dispositivo.

La luz LED tiene dos funciones. La primera es durante el proceso de asociación a la red WiFi, donde parpadea de acuerdo a como se indico en el Capítulo 2 para indicar en qué estado se encuentra el proceso. Luego, una vez conectado a la infraestructura WiFi indica el estado de la salida del actuador.

3.3.2 Rutina de funcionamiento

Las tareas que realiza el equipo periférico pueden clasificarse de forma general en dos tipos. Las funciones de asociación a la red WiFi, las cuales corresponden a las de un dispositivo consultante. Estas se describieron en el Capítulo 2. El otro tipo de tareas son las que integran el funcionamiento de actuador en el sistema domótico. Estas comienzan a partir del momento que se conecta a la red WiFi anfitriona y se describirán en esta sección.

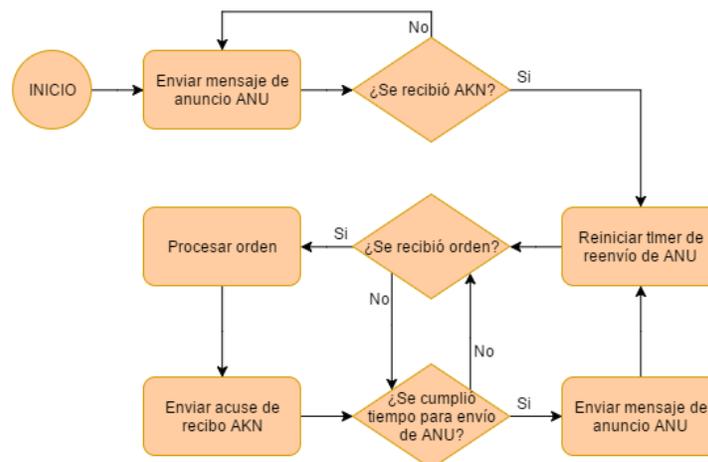


Figura 3.20 Diagrama de Flujo de rutinas del periférico

Luego de conectarse a la red WiFi, el periférico debe anunciarse a la interfaz de usuario para que esta pueda registrarlo y posteriormente interactuar con él. Luego de enviar el mensaje ANU se aguardara por un tiempo. Si no se recibe el AKN de la interfaz de usuario, se volverá a enviar. Este proceso se repetirá hasta recibir la confirmación de recepción.

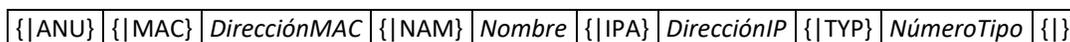


Figura 3.21 Trama de mensaje de anuncio que envía el periférico

Cuando el periférico recibe el mensaje de AKN, inicia un timer el cual, una vez que indique que se cumplió un determinado tiempo, se volverá a enviar un mensaje de anuncio a la interfaz de usuario para mantener actualizado el estado.

A partir de ese momento el periférico entra en un bucle donde se revisan dos cosas. Primero si se recibió una orden, y segundo si ya se cumplió el tiempo del timer para volver a enviar un mensaje de ANU.

Si se recibió una orden, el periférico procede a interpretarla y coloca el estado de la luz LED según corresponda.

Lexema	Función
{ SOH }	Encender
{ SOL }	Apagar

Tabla 3.3 Órdenes que puede recibir el periférico

Estas órdenes no poseen remitente, porque en el esquema sobre el cual funciona este prototipo, solo hay un dispositivo del cual pueden recibir mensajes, que es la interfaz de usuario. Originalmente el mensaje sale del puerto serie de la placa Arduino con la dirección IP incluida del periférico al cual va dirigido. Pero este campo es removido por el ESP8266 de la interfaz de usuario una vez que extrae la dirección IP y sabe a dónde debe enviarlo. Por eso a los periféricos solo le llega el lexema que indica la orden.

Luego de procesar la orden el periférico envía un mensaje de AKN para que se registre el suceso de la operación. Este mensaje contiene la dirección MAC de este, para que la interfaz de usuario pueda identificar de qué periférico proviene la notificación.



Figura 3.22 Estructura de la trama de confirmación de recepción que envía el periférico

En cada bucle de la rutina se verifica cuanto tiempo pasó desde la última vez que se envió un mensaje de ANU. Cuando se detecta que pasó el tiempo pre-asignado para esa función, el periférico procede con el reenvío de ese mensaje. En este caso no se espera el acuse de recibo de la interfaz de usuario. Esto es porque si en un principio la interfaz registró al periférico, esta guarda la información en la memoria no volátil. Entonces incluso a pesar de que esta se apague no necesitara registrarlo de nuevo para enviarle mensajes. Cuando la interfaz de usuario inicie luego de un periodo de estar apagada, considerará que el periférico estará online y el usuario podrá operar sin pasar por ningún periodo de desconexión.

En caso que la dirección IP del periférico cambie mientras la interfaz estuvo apagada, o por algún error no está registrado en la memoria no volátil, a lo sumo habrá que esperar el tiempo de reenvío de ANU en el periférico para que la interfaz de usuario actualice la información.

3.4 Conclusiones

En este Capítulo se describió con detalle la implementación del prototipo del sistema. Por su complejidad, se le ha dedicado la mayor parte de la sección a la interfaz de usuario. La misma coordina el módulo de reconocimiento de voz y recibe mensajes de la interfaz WiFi. A su vez el microcontrolador lleva a cabo diferentes procesos internos para la gestión de los periféricos, para los cuales sus recursos fueron distribuidos eficazmente. Esto se pudo comprobar con su posterior uso en diferentes instancias.

En cuanto a los dos periféricos, fueron desarrollados íntegramente con un módulo ESP8266 cada uno, de esta forma se optimizaron los recursos para el proyecto. El poder de procesamiento del microcontrolador integrado en el chip ESP8266EX cumple con las exigencias de la rutina y le sobran recursos para agregar más funciones en futuras implementaciones.

El mayor problema que ha surgido es el que producen las interferencias de transmisores de radiofrecuencias próximos a la interfaz de usuario. La presencia a menos de 50cm puede ocasionar el bloqueo de la capacidad de identificar los comandos de voz. Pero estas circunstancias de funcionamiento no son las habituales. Por lo tanto este problema puede pasar inadvertido durante un uso normal.

El funcionamiento general del sistema es aceptable y se han cumplido con los objetivos designados para el prototipo.

CAPÍTULO 4

4.1 Discusión de los resultados

El objetivo que se planteó en este desarrollo fue el de crear un sistema que pueda asistir a personas con discapacidades motrices para utilizar de forma autónoma determinados elementos en el hogar. Al reducir el esfuerzo físico para accionarlos se aumenta la independencia del usuario y su calidad de vida.

Para minimizar la interacción física del usuario con el sistema se ha propuesto la realización de una interfaz de manos libres. Además la instalación del sistema debe ser sencilla y permitir la expansión del sistema sin mayores modificaciones. También el proyecto está orientado a que posea un costo final accesible y que el producto posea características competitivas para la comercialización.

Las tecnologías que se escogieron para lograr estos requisitos fueron el reconocimiento de voz y la conexión vía WiFi. La primera logra utilizarse como interfaz manos libres mediante un módulo configurable, el cual reconoce los comandos sin la necesidad de estar conectado a internet. Mientras que la conexión inalámbrica permite que el sistema sea fácil de integrar al hogar a la vez que establece un límite razonable para la expansión del sistema, ya que habilita la conexión de más de 200 equipos en red.

La capacidad inalámbrica de WiFi simplifica la inserción física del sistema en el hogar, pero poner en funcionamiento gran cantidad de dispositivos se puede volver una tarea tediosa. Si no se cuenta con la capacidad de asociar mediante WPS, el usuario debe elegir la red WiFi anfitriona bajo la cual funciona el sistema y suministrarle la contraseña de seguridad a cada equipo para que se conecte. Por esto, el solo hecho de usar esta tecnología no hace que el sistema sea de fácil instalación y expansión. Es necesario contar una plataforma que permita dar soporte a las funcionalidades planteadas.

Durante el desarrollo del proyecto se ha hecho hincapié en que el sistema debe ser modular, descentralizado y escalable. Siguiendo estos lineamientos, se desarrolló un protocolo de asociación a la red WiFi el cual soluciona el inconveniente que surge al querer asociar muchos dispositivos en la red. Se logró que proceso requiera la intervención del usuario o instalador solo para conectar el primer equipo. El resto de los nodos se conectan a partir de este, el cual es considerado como el dispositivo de referencia. Esto simplifica la instalación, y da soporte a la escalabilidad, ya que se realiza el mismo esfuerzo para cualquiera que sea la cantidad de dispositivos que se deseen asociar.

También este protocolo hace que no haya diferencias entre realizar la instalación de una sola vez o ir agregando equipos con el tiempo, ya que solo requiere una configuración inicial. Esto, junto con la modularidad, permite que el usuario adapte el sistema a sus necesidades actuales y lo incentiva a que lo pueda ir mejorando si sus necesidades cambian. Esto último le da competitividad al producto en el mercado.

Como el funcionamiento del sistema es dependiente del dispositivo de referencia, pueden surgir problemas en el caso de que este no se encuentre disponible. Para mejorar la robustez de este esquema ante una situación de este tipo, se ha establecido que esta función de referencia sea rotativa entre todos los dispositivos que están conectados y sea elegible por el usuario. Esto quiere decir que no hay un dispositivo diseñado exclusivamente para ese funcionamiento, sino que todos son compatibles. Puede seleccionarse indistintamente cualquiera que se vaya a utilizar en la red para que desempeñe esa función.

Para el desarrollo de este protocolo de asociación a WiFi se necesita que cada equipo en la red posea un microcontrolador con el poder de procesamiento necesario. Por eso se utilizaron los módulos ESP8266, los cuales no solo cumplen con los requisitos, sino también que poseen un costo muy accesible.

El siguiente paso después de establecer el protocolo de asociación a WiFi fue crear un protocolo de domótica. El mismo se basó en un enfoque IoT (*Internet of Things*) el cual consiste en tomar cada dispositivo como una abstracción del objeto físico real en la red. Esto conduce a un sistema descentralizado en el cual puede haber varios agentes que interactúen con los diferentes elementos. Para funcionar así no es

estrictamente necesario que sea a través de internet. Lo primordial es utilizar los protocolos TCP/IP. El módulo ESP8266 se ajusta perfectamente a este tipo de esquemas ya que posee la pila TCP/IP integrada. De esta forma se ha trabajado sobre la capa de aplicación en el modelo TCP/IP y se ha creado un lenguaje de comunicación de máquina a máquina basado en la extracción de datos particulares de una cadena de caracteres. La extracción se hace mediante un software *parser* que se diseñó especialmente para este proyecto.

El protocolo de domótica que se desarrolló se basa en que cada dispositivo en la red posee una serie de características, las cuales puede compartir con cualquier otro que se lo solicite. Entre estas se encuentra el tipo de dispositivo. Esto se refiere a si se trata de un actuador o sensor, si tiene dos estados o múltiples estados, si es una interfaz de usuario, etc. El tipo define los comandos que el dispositivo puede procesar. A su vez, cada dispositivo posee una programación la cual realizara diferentes tareas dependiendo del tipo que posea el par con el que va a interactuar. Este funcionamiento se apoya en que los dispositivos sean capaces de hallar a los otros en la red y comunicarse con ellos. Esto fue un problema ya que la dirección de cada uno es variable y no se conoce de antemano. Para solucionar esta dificultad, se programó una rutina que permite que el dispositivo de referencia lleve un registro de todas las direcciones de los equipos que se van asociando al sistema. Entonces, la función del dispositivo de referencia es primordial, no sólo en el protocolo de asociación, sino también en el de domótica. Ya que la dirección de este es siempre la misma, y el resto de los equipos lo usarán para consultarle por las direcciones de sus pares en cualquier momento.

Así, se sentaron las bases para un sistema modular que se puede adaptar a diferentes escalas de aplicación. Sin embargo no se pudo conseguir una descentralización completa ya que siempre se necesita de un equipo de referencia. A pesar de esto se puede decir que tiene dos ventajas sobre un sistema centralizado clásico. La primera es que esta función es rotativa o intercambiable con los otros equipos que forman el sistema, lo cual le da robustez. Y la segunda es que solo es necesario para el descubrimiento de otros equipos, no para mantener la comunicación entre estos. Esto permite que, en el caso de que no esté disponible el equipo de referencia, los equipos que ya se encuentran identificados sigan funcionando hasta que se cambie la dirección de red que poseen.

En la implementación del prototipo no se pudieron adaptar todas las características del concepto inicial. Esto es debido a falta de recursos en esta instancia del proyecto para llevar a cabo la complejidad propuesta. El prototipo se enfocó en la accesibilidad de la interfaz de usuario y al sistema de comunicación entre los módulos. Los sensores y actuadores que intervienen en los objetos o variables del hogar quedaron fuera de este desarrollo. Se consideró que el principal desafío para lograr los objetivos propuestos es cómo trasladar y recibir los comandos a cada estación periférica. La traducción del comando en la acción se puede adaptar mediante soluciones de sensores y actuadores que ya hay disponibles en el mercado. Tampoco se pudo implementar la rotatividad del dispositivo de referencia. Esto debido a limitaciones de hardware del modelo ESP-01 del módulo ESP8266 que se utilizó. Lo cual se puede solucionar implementando los equipos con otro modelo de módulo, por ejemplo el ESP-12 que posee más pines GPIO.

Si bien el protocolo es compatible con varias interfaces de usuario en simultáneo y con periféricos de múltiples estados, debido a las limitaciones antes mencionadas, esto tampoco se pudo implementar. El prototipo desarrollado consta de una unidad de interfaz de usuario, la cual cuenta con reconocimiento de voz y dos dispositivos periféricos actuadores de dos estados, que encienden o apagan una luz LED a partir de los comandos que se le envían. El sistema se probó en diferentes lugares de residencia. En las pruebas se detectó una falla en el funcionamiento que consiste en el bloqueo de la identificación de los comandos de voz. El mismo se produce por la cercanía de más de 50cm de algún equipo emisor de radiofrecuencias. Esto ha sucedido en condiciones de uso especiales, ya que la interfaz de usuario se encuentra a más de 50cm durante el funcionamiento normal. Respetando esta restricción, se repitieron las pruebas y el sistema

funcionó de la forma esperada. La interfaz de usuario registra de forma efectiva los comandos de voz para navegar por un menú que va actualizando de forma dinámica. Los periféricos reciben las órdenes y proceden con la ejecución de forma correcta. Cuando un equipo se desconecta, luego se reincorpora de la forma prevista a la red. El procedimiento de asociación a la red se ha registrado exitoso en todas las situaciones de prueba.

4.2 Conclusiones finales

Los principales objetivos que se propusieron se lograron cumplir satisfactoriamente. La interfaz se puede controlar de forma eficaz mediante los comandos de voz, lo cual minimiza el esfuerzo físico para operar los elementos que están conectados al sistema. Esto resulta de ayuda no solo a personas con discapacidades motrices, sino a cualquier tipo de usuario. La comunicación de los equipos funciona con una red WiFi anfitriona, pero sin necesitar conexión a internet. Tampoco la necesita el módulo de reconocimiento de voz, a diferencia de los servicios más usados para esta función. Al no depender de proveedores de internet aumenta la confiabilidad del sistema. El sistema WPS y el protocolo de asociación a la red WiFi desarrollado ofrecen un mecanismo sencillo para la instalación del sistema y su posterior expansión. Por lo antes descrito, el desarrollo se adapta a las necesidades del usuario.

El desarrollo de la interfaz de usuario se pudo llevar a un punto donde es factible de utilizar en cualquier hogar. Los periféricos también pueden usarse agregando los circuitos necesarios para interactuar como interruptores de equipos instalados en el hogar. También se podrían desarrollar nuevos periféricos para adaptarse a las necesidades particulares del usuario.

En cuanto a los objetivos secundarios, se logró un bajo costo del sistema. Esto es debido a la reutilización de los módulos ESP8266. Gracias a su capacidad de programación y hardware integrado, no solo funcionan como interfaz WiFi, sino también como microcontroladores. La mayoría de los periféricos pueden realizarse con un solo módulo ESP. Esto economiza significativamente la implementación. Solo en los casos que se necesite coordinar algún módulo particular es preciso el uso de algún microcontrolador adicional. Como fue en el caso del módulo reconocedor de voz y el display en la interfaz de usuario, para lo cual se usó una placa Arduino. Se considera que para darle competitividad comercial al producto frente a otros sistemas se deben desarrollar aspectos que no se exploraron en el prototipo, como lo es la automatización, el acceso remoto y la diversidad de periféricos para diferentes usos.

4.3 Mejoras del proyecto

Durante el desarrollo del proyecto, algunas de las características que surgieron de su concepción básica quedaron sin realizar porque se priorizó la accesibilidad y comunicación del sistema. A su vez, se creó un protocolo de domótica que planteó características que excedían los objetivos del proyecto. Estas funciones quedan propuestas a continuación para futuras mejoras del proyecto.

- **Nuevos periféricos:** se podrían implementar periféricos con más funciones, como sensores y actuadores de múltiples estados para poder aprovechar al máximo las capacidades de la plataforma desarrollada.
- **Automatización:** este es un aspecto básico de la domótica que no se ha desarrollado. La automatización permite optimizar los recursos en el hogar y brinda más confort ya que las operaciones se realizan sin la intervención del usuario.

- **Aumentar cantidad máxima de periféricos:** este prototipo se preparó para controlar hasta 8 dispositivos. Esto debido a que se decidió no paginar el menú y es el máximo que entra en el tamaño de la pantalla del display usado. Se puede implementar otro menú mediante otras formas de visualización y/o paginado. De esta forma se aprovecharía la capacidad de conectar más de 200 equipos que provee WiFi.
- **Ahorro de energía:** Los dispositivos ESP8266 poseen sofisticadas características de ahorro de energía que no se aprovecharon en este prototipo. Esto es relevante cuando el equipo funciona por medio de baterías.
- **Implementar la rotatividad del dispositivo de referencia:** esta capacidad fue pensada en el concepto inicial de los protocolos desarrollados. Agregar esta función aumenta la robustez del sistema. Para lograrlo sería necesario modificar las rutinas que maneja cada equipo y contar con un GPIO adicional en los módulos ESP8266 para poder agregar un interruptor de selección.
- **Compatibilizar el prototipo para utilizar varias interfaces de usuario en simultáneo:** si se implementa el punto anterior es fácil agregar varias interfaces de usuario para que funcionen al mismo tiempo. Esto debido a que la rutina original de la interfaz de usuario debe modificarse para que consulte los datos al dispositivo de referencia y por lo tanto es lo mismo si el sistema opera con una o varias interfaces de usuario.
- **Aplicación para asociar:** para utilizar el protocolo de asociación se necesita un software de terceros para conectarse al equipo de referencia y brindarle los datos necesarios. Se puede desarrollar una aplicación que guíe al usuario en los pasos.
- **Comandos de voz personalizados:** se puede agregar la capacidad de asignarle a cada opción del menú un comando de voz dependiente del locutor. De esta forma el usuario podría asignarle a cada periférico un comando de voz personalizado mediante un proceso de entrenamiento.
- **Conectividad a internet:** Los módulos ESP están preparados para conectarse directamente a internet. Si el enrutador utilizado como infraestructura WiFi también tiene conexión a internet se puede utilizar algún servidor en la nube como interfaz de usuario para lograr acceso remoto al sistema. Incluso también se puede utilizar como dispositivo de referencia.
- **Seguridad:** cada dispositivo responde a los comandos siempre y cuando estos le lleguen con el formato adecuado. No tiene mecanismos de validación del emisor de la comunicación. Esto genera una vulnerabilidad en el sistema. Se considera necesario un sistema de autenticación si se implementa la conectividad a internet, ya que se expone a mayores riesgos. Una solución a esto puede ser generar algún protocolo que permita asignar un grupo de pertenencia a los dispositivos. De esta forma los equipos solo responderán comandos a los pares tengan la misma identidad de grupo.

Bibliografía

- [1] Colegio de Ingenieros Especialistas de Córdoba, 2012, *“Domótica Registrada. Guía de contenidos mínimos para la elaboración de un proyecto de domótica”*
<http://www.ciec.com.ar/images/archivos/Domotica-CIEC.pdf>
- [2] Dave Rye, 1999, *“My life at X10”*
<http://www.hometoys.com/content.php?url=/htinews/oct99/articles/rye/rye.htm>
- [3] Dave Rye, *“Technical Note”*
http://jvde.us/info/x10_protocol.pdf
- [4] KNX Association, 2016, *“KNX ya es Estándar Internacional: ISO/IEC 14543-3”*
<http://www.knx.org/fileadmin/news/120524715414725PR20061201-ES.pdf>
- [5] Hubert Zimmermann, 1980, *“OSI Reference Model. The ISO Model of Architecture for Open Systems Interconnection”*
http://web.archive.org/web/20050309080952/http://www.comsoc.org/livepubs/50_journals/pdf/RightsManagement_eid=136833.pdf
- [6] KNX Association, 2016, *“Conocimientos básicos del estándar KNX”*
https://www.knx.org/media/docs/downloads/Marketing/Flyers/KNX-Basics/KNX-Basics_es.pdf
- [7] Dave Evans, 2011, *“Internet de las cosas. Cómo la próxima evolución de Internet lo cambia todo”*
http://www.cisco.com/c/dam/global/es_mx/solutions/executive/assets/pdf/internet-of-things-iot-ibsg.pdf
- [8] Bluetooth Website, 2016, *“Bluetooth technology basics”*
<https://www.bluetooth.com/what-is-bluetooth-technology/bluetooth-technology-basics>
- [9] P.Rohitha, P. Ranjeet Kumar, N. Adinarayana, T. Venkat Narayana Rao, 2012, *“Wireless Networking Through ZigBee Technology”*
https://www.ijarcse.com/docs/papers/July2012/Volume_2_issue_7/V2I700137.pdf
- [10] IEEE, 2016, *“802.15.4q-2016 - IEEE Standard for Low-Rate Wireless”*
<http://standards.ieee.org/findstds/standard/802.15.4q-2016.html>
- [11] Rabbit, 2008, *“An Introduction to Wi-Fi”*
http://ftp1.digi.com/support/documentation/0190170_b.pdf
- [12] Elechouse Website, 2016, *“VR3”*
<http://www.elechouse.com/elechouse/images/product/VR3/>
- [13] Elechouse Website, 2016, *“SimpleVR”*
<http://www.elechouse.com/elechouse/images/product/simpleVR/>

[14] VeevaR, 2016, "*EasyVR 3 User Manual Release 1.0.14*"
<http://www.veear.eu/files/EasyVR%203%20User%20Manual%201.0.14.pdf>

[15] Google Cloud Plataform, 2016, "*Speech API*"
<https://cloud.google.com/speech/>

[16]
Forouzan B.A. TCP/IP "*Protocol Suite*" (4) New York: Mc Graw Hill

[17] Espressif Website, 2016, "*Documents*"
<https://espressif.com/en/support/download/documents>

[18] Arduino Website, 2016, "*Arduino Software (IDE)*"
<https://www.arduino.cc/en/Guide/Environment>

[19] ESP8266 Community Forum, 2016, "*ESP8266 core for Arduino*"
<https://github.com/esp8266/Arduino>

[20] Eduardo Serna-Pérez, 2010, "*Análisis Léxico*"
<http://www.paginasprodigy.com/edserna/cursos/compilador/notas/Notas2.pdf>

[21] Arduino Website, 2016, "*Arduino Mega Board*"
<https://www.arduino.cc/en/Main/arduinoBoardMega>

[22] Texas Instruments, 2014, "*Logic Guide*"
<http://www.ti.com/lit/sg/sdyu001aa/sdyu001aa.pdf>



RINFI se desarrolla en forma conjunta entre el INTEMA y la Biblioteca de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata.

Tiene como objetivo recopilar, organizar, gestionar, difundir y preservar documentos digitales en Ingeniería, Ciencia y Tecnología de Materiales y Ciencias Afines.

A través del Acceso Abierto, se pretende aumentar la visibilidad y el impacto de los resultados de la investigación, asumiendo las políticas y cumpliendo con los protocolos y estándares internacionales para la interoperabilidad entre repositorios



Esta obra está bajo una [Licencia Creative Commons Atribución-NoComercial-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/).