

Proyecto 2: Seguimiento del usuario con reproducción musical.

Sistema de música Wall-E

Carlos Andrés Solano Artavia 2013044781

andressolano0508@gmail.com

Juan Pablo Ortiz Jarquin 201218772

ortizjp2003@gmail.com

Jill Carranza Otárola 2013101387

jillcarranza@gmail.com

Randall Bonilla Vargas 2013018028

randallmbv@gmail.com

Abstract—The following document shows the development and analysis for the design, implementation and creation of a music follower system, by means of bluetooth, giving a new experience to the user, where he can configure through an application his music player device to his preference in any space in the home..

Resumen—El siguiente documento muestra el desarrollo y análisis para el diseño, implementación y creación de un sistema seguidor de música, mediante bluetooth, dando una experiencia nueva al usuario, donde este puede configurar mediante una aplicación su dispositivo reproductor de música a su preferencia en cualquier espacio del hogar.

dadas mediante un app donde se puede escoger el modo de preferencia., dando una mejor experiencia a la persona sea donde sea que este se desplace en su hogar. En este documento se encuentra toda la información pertinente al desarrollo, proceso e implementación de la solución, resultados y retos encontrados a lo largo del desarrollo del proyecto.

I. INTRODUCCIÓN

Hoy en día las aplicaciones en el área de domótica son de gran interés, por la accesibilidad, agilidad y flexibilidad que esta brinda a las personas en su día a día, es por esto que en el presente proyecto se tiene como objetivo la creación de un sistema seguidor de música, el cual reproduce la música inalámbricamente desde el celular o un dispositivo con bluetooth, a distintas áreas de un hogar que cumpla con especificaciones

II. MARCO TEÓRICO

Para un entendimiento acertado de la solución dada, es necesario conocer conceptos básicos y algunas consideraciones previas, estas se explican brevemente en esta sección.

II-A. Ambiente computacional y de trabajo.

Los sistema embebidos son dispositivos que procesan datos digitalmente y están diseñados para una función específica, estos usan microcontroladores, FPGAs o DSPs, estos están optimizados para mejorar tamaño costo, consumo confiabilidad y desempeño. según Simón (2013) los sistemas embebidos continúan incrementando su número y complejidad, hoy en día se reconoce que el manejo explícito de la dimensión temporal en el software es un concepto fundamental que debe introducirse en forma

temprana en los estudiantes, donde se introducen algunos conocimientos principalmente en torno a la idea de utilizar un microcontrolador como plataforma experimental.

Los sistemas embebidos simples o de menor complejidad se basan en microcontroladores, que incluyen en un único chip, el procesador, memoria, interfaces, conversores, timers, etc. En particular, la plataforma open source Arduino, aparte de su bajo costo expone a los estudiantes suficiente complejidad y desafíos propios de la programación embebida. Cruz (2012)

Existen importantes protocolos de comunicación según menciona Acosta (2010) los cuales son importantes de conocer y detallamos a continuación.

- Cuando se desea transmitir un conjunto de datos desde un componente electrónico a otro, se puede hacer de múltiples formas:
 - Comunicación Serie: la información es transmitida bit a bit (uno tras otro) por un único canal.
 - Comunicación Paralela: se envían varios bits simultáneamente, cada uno por un canal separado y sincronizado con el resto.
- I²C (Inter-Integrated Circuit): Conocido como TWI (Two-wire), es un sistema utilizado para comunicar circuitos integrados entre sí, necesita de dos líneas para transmitir la información: SDA: transfiere los datos SCL: envía la señal de reloj.
- SPI (Serial Peripheral Interface): Sistema que permite controlar (a cortas distancias) casi cualquier dispositivo electrónico digital que acepte un flujo de bits serie sincronizados (regulados por un reloj)

El sistema embebido a utilizar en este proyecto es el Arduino Uno R3, pues cumple con los requerimientos suficientes para un desarrollo adecuado del mismo, además de facilitar la utilización de sus recursos, simplifica

enormemente la programación sobre él, pues cuenta con una herramienta editora propia de la marca con extensas librerías y ejemplos dados por el fabricante, su aspecto físico se muestra a continuación.



Figura 1. Arduino Uno R3

El voltaje de funcionamiento de la placa Arduino (incluyendo el microcontrolador y el resto de componentes) es de 5V, la placa Arduino puede conectarse a la PC mediante cable USB, la alimentación recibida será regulada permanentemente a los 5V de trabajo y ofrece un máximo de hasta 500mA de corriente (potencia consumida de unos 2,5 W)

Dispone de 14 pines de E/S digitales, numeradas desde 0 hasta la 13, aquí se pueden conectar sensores para que la placa pueda recibir datos del entorno o actuadores para que la placa pueda enviarles las órdenes pertinentes. Todos los pines funcionan a 5V y pueden proveer o recibir un máximo de 40mA. Por su parte también dispone de 6 pines de entradas analógicas, pueden recibir voltajes continuos de 0V a 5V.

Para este proyecto los pines Rx y Tx, son de suma importancia pues permiten la comunicación sin intermediarios de dispositivos externos, recibir directamente datos en serie (Rx) o transmitirlos (Tx) desde el módulo bluetooth.

El Software Arduino es Integrated Development Environment (IDE) Entorno de

Desarrollo Integrado. En el caso de Arduino, se necesita un IDE que nos permita escribir y editar el programa (llamado “sketch”), comprobar si hay errores, si es correcto para grabarlo en la memoria del microcontrolador

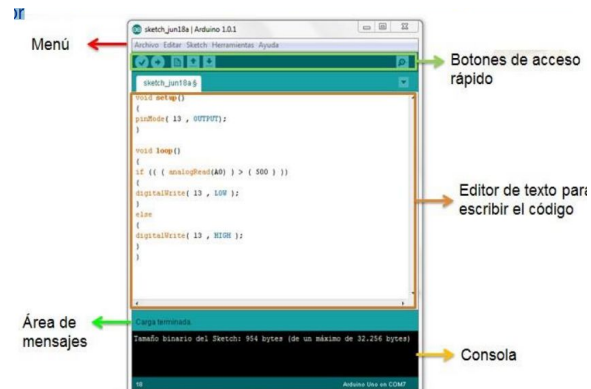


Figura 2. Software arduino

Las desventajas o limitante de este sistema embebido se mencionan en la sección de Limitaciones y Recomendaciones.

Se procede a mencionar los componentes utilizados a su vez se detalla el objetivo y función que cada componente realiza en el desarrollo del proyecto.

Inicialmente, por los requerimientos del proyecto, se sabe que es necesario la utilización de un módulo bluetooth, pues la música / audio debe transmitir inalámbricamente. Como primera opción se pensó en el módulo bluetooth HC-05 para transmisión de audio, sin embargo este módulo es incapaz de realizar dicha función, pues en sus funciones como módulo no se contempla la transmisión de audio, únicamente se permite la transmisión serial. Es por esto que se utiliza para la transmisión del audio un módulo bluetooth estéreo XS3868.



Figura 3. Módulo Bluetooth Estéreo XS3868.

Una de las desventajas de este módulo stereo es que no se puede utilizar para enviar valores o instrucciones para acciones dentro del programa de arduino, es por ello que de igual forma se utiliza el módulo Bluetooth HC-05. En resumen, se utiliza el módulo XS3868 para la transmisión de audio / música y el módulo HC-05 para el control del programa mediante la aplicación en el dispositivo.



Figura 4. Módulo Bluetooth HC-05.

Tomando en cuenta las especificaciones del proyecto, se debe utilizar un sensor que permita detectar la presencia de personas en las habitaciones, de modo que al existir presencia de personas la aplicación realice una acción al respecto, en este caso, que se active el sistema de audio, si bien es cierto se pueden usar sensores de movimiento, de calor, de área, sin embargo por su facilidad

y comodidad el sensor utilizado es el HC SR04, un sensor ultrasónico el cual permite determinar la distancia de un objeto desde la placa. Para ello se utilizará la librería de arduino especial para ultrasónico.



Figura 5. Sensor Ultrasónico HC SR04.

Se debe recalcar que se utilizan estos sensores pues cumplen los requerimientos suficientes para la correcta comprobación de concepto y funcionamiento del mismo, para un caso más real se debe utilizar un sensor que determine la posición o cercanía de un dispositivo ya sea celular, tablet entre otras, que contenga la aplicación vinculada.

Seguidamente se debía diseñar el modo o tipo de acción de control para que el sistema permitiera o no la reproducción de música en la habitación indicada. Es por ello que se decidió interrumpir la señal de audio directamente, de esta forma, la señal de control debe permitir o no el paso de dicha señal de audio, esto se logra con la utilización de un relé de 4 canales (pues solo se utilizará dos parlantes) compatible con arduino, que permite el paso mediante una señal de control proveniente del arduino. El relé se observa físicamente de la siguiente manera.

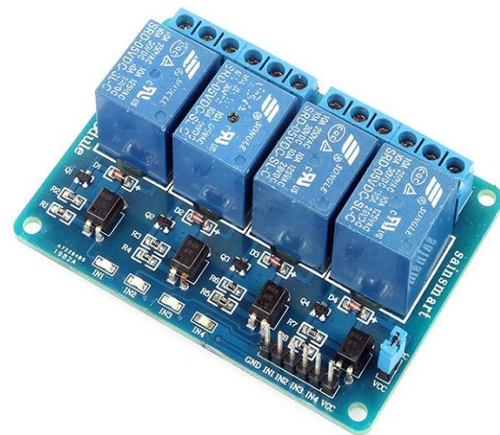


Figura 6. Relé de 4 canales 5 Voltios.

Por otra parte se hizo uso de una pantalla LCD la cual, muestra en pantalla los diferentes modos de configuración del sistema seguidor de música, los cuales se detallaran en la siguiente sección.

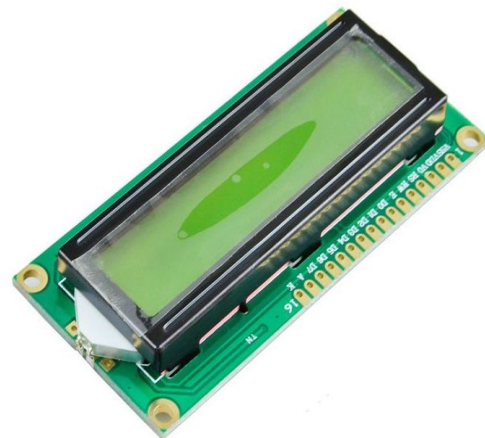


Figura 7. Pantalla LCD

Cabe destacar que para la implementación de este proyecto se requirió de demás componentes mostrados en la siguiente figura.

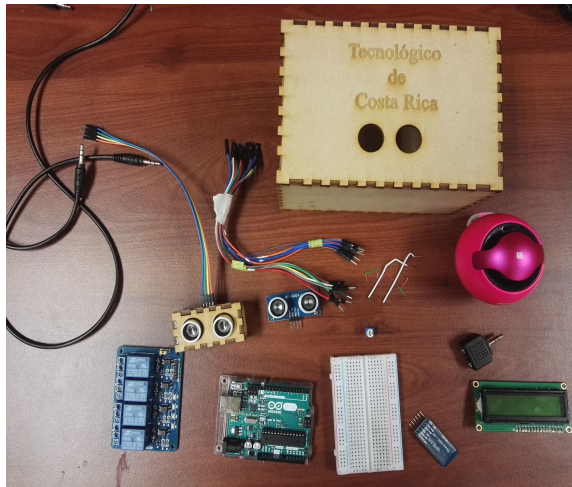


Figura 8. Demás componentes utilizados

Se requirió, una mini protoboard, un potenciómetro, jumpers macho-macho, así como macho- hembra, puentes, cables tipo jack, así como un conector adaptador audio estéreo mini jack macho 3.5 mm a doble jack hembra 2471, parlantes con conexión jack macho, y por último una caja donde se termina de montar todo el sistema.

III. SOLUCIÓN IMPLEMENTADA

Para la solución implementada, se procede a separar en bloques o secciones de trabajo para desarrollar de forma ordenada y clara la solución.

III-A. Descripción de la solución.

Para la implementación de la solución se define una estructura para el programa, de forma que permita optimizar y minimizar el procedimiento, utilizando la herramienta y editor de arduino pues ese fue el sistema embebido escogido para implementar la solución, pues cuenta con los recursos suficientes y necesarios para desarrollar una solución adecuada.

El programa inicia con la inclusión o llamado de algunas librerías necesarias para el proyecto, como lo es “Ultrasonic”, librería que realiza la comunicación y obtención de datos del sensor ultrasónico. Además se incluye la librería para la LCD, que facilita el control de la

misma al desarrollador. Posterior a esto se declaran las variables necesarias para la ejecución y desarrollo del programa, seguido de la inicialización de la LCD, del puerto serial y declaración de salidas en los pines del arduino.

Una vez declaradas las variables globales e inicializado la comunicación con los periféricos (en el void loop) se procede a desarrollar el programa que permite la aplicación en sí, esto se hace en el void loop, el cual es un ciclo que se repite infinitamente ejecutando una única aplicación, es decir, la placa arduino solo permite la ejecución de una aplicación / programa en él. En esta sección del programa, inicialmente se realiza la comunicación con el módulo Bluetooth HC-05 de la siguiente forma.

Se lee el contenido en el puerto serial.
Se asigna el valor leído a la variable estado.
Dependiendo del valor de “estado” se le asigna un valor a la variable x.
La variable x controla el flujo de ejecución del programa.

La variable estado corresponde al modo que el usuario desea, ya sea modo Seguir, Fiesta o Home. Estos modos cumplen las siguientes funciones:

1. Seguir: El dispositivo detecta la presencia de una persona y activa el sistema de audio en esa habitación. De esta forma el sistema activa únicamente el sistema de audio donde se encuentre la persona (solo un parlante a la vez).
2. Home: El sistema activa todo los parlantes de la casa, es decir, funciona como un sistema de audio convencional.
3. Fiesta: El sistema detecta la presencia de personas y enciende el audio, a diferencia del modo seguir, este modo permite la reproducción en múltiples habitaciones, es decir habrá música

donde haya personas (uno o más parlantes a la vez).

Por otro lado la variable x, permite entrar a la sección de programa correspondiente para cada modo, de esta forma se dirige o controla el flujo de ejecución del programa. Cabe resaltar que por decisión de los desarrolladores, se decide utilizar la función while para separar cada modo, pues permite un mejor control y facilita el flujo de ejecución.

Se procede a explicar la sección de programa para cada modo. Inicialmente si estado es igual a 3, x toma el valor de 1, de esta forma se cumple la condición del primer while, que es el del modo party. Una vez dentro de este while se pinta en pantalla (LCD) el modo en el que se está, para que de este modo el usuario tenga una mejor visualización del correcto funcionamiento de la app.

Posterior a esto se lee el ultrasónico y se guarda en la variable sonar1 y sonar2 el valor numérico en centímetros de la distancia de algún objeto (persona) a el sensor, de esta forma se ajusta esa distancia para activar o desactivar las señales de control que encienden o apagan el sistema de audio.

Seguidamente se cuenta con cuatro condicionales (la cantidad de condicionales (if) varía de acuerdo al modo). El primer if se ejecuta si ambos ultrasónicos cumplen con el mínimo de distancia requerido para activar el sonido, además de la condición siga, que se utiliza como auxiliar para el control del flujo de ejecución del programa en este caso se le asigna un valor de HIGH a ambas salidas del arduino, activando el sistema de audio en ambas habitaciones. El segundo if (else if) se ejecuta si el sonar1 cumple los requisitos, se activa una sola señal de control, así mismo ocurre lo mismo en el tercer caso, solo que esta vez para el segundo sensor. Finalmente el cuarto caso asigna un LOW a ambas señales de control, de esta forma en caso de que el sistema esté encendido pero no detecta la presencia de personas en las habitaciones, no habrá sonido.

Para el segundo modo, el modo Home, se simplifica el proceso, pues simplemente se asigna un HIGH a ambos parlantes pues, como ya se mencionó, en este modo el sistema actúa como un sistema de audio convencional. En el caso del modo seguir, se da un caso similar al del modo Fiesta, con la diferencia de que no se activan más de un parlante al mismo tiempo. Con el objetivo de dar una mejor visualización de lo anteriormente mencionado, se muestra el siguiente diagrama de flujo.

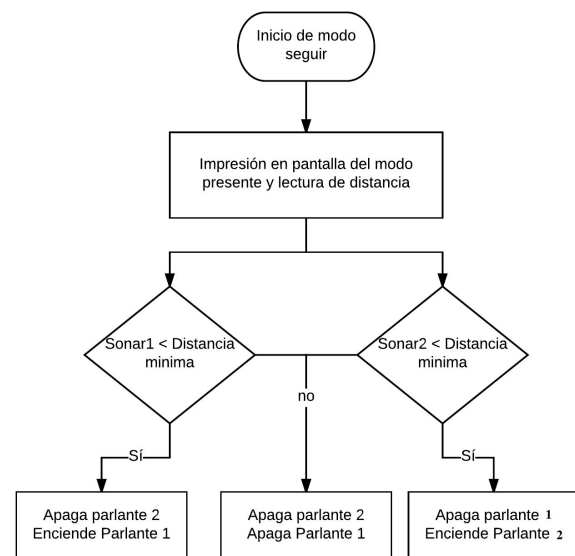


Figura 9. Diagrama de flujo del funcionamiento del modo seguir.

Antes de proseguir con la explicación, se debe mencionar el funcionamiento de la etapa de control del sistema de audio. Como ya se mencionó, dependiendo del modo, se asignan valores HIGH o LOW para las salidas del arduino, siendo estas salidas señales de control para dicha etapa de control. La etapa de control se basa en una placa con un conjunto de relés, los cuales permiten el paso de la señal de audio sí y sólo si, la señal de control se los permite. De esta forma cada modo, envía un par de señales de control que habilitan o no el paso de señal de audio para el parlante.

Para el control de los modos a escoger se utiliza una app, la cual fue desarrollada mediante la herramienta MIT App Inventor 2, la cual facilita enormemente la creación de apps.

Esta herramienta se basa en la programación por bloques, de esta forma la aplicación para esta aplicación en desarrollo se observa de la siguiente forma.

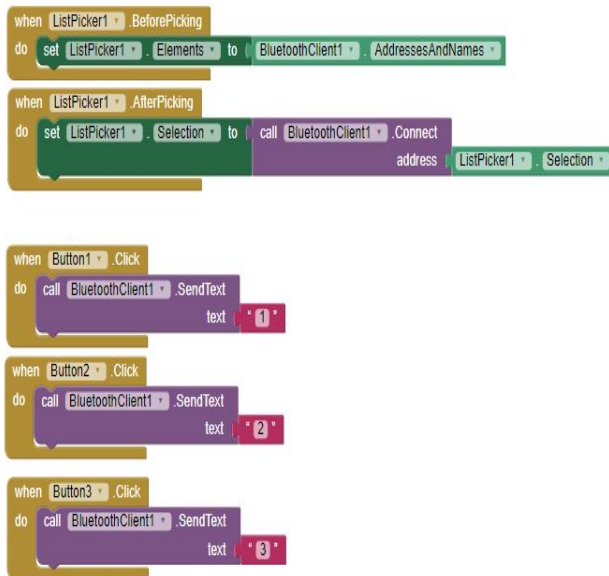


Figura 10. Bloques de programación para App.

Además la herramienta del MIT App Inventor 2, permite el diseño de la interfaz gráfica de la aplicación, la misma fue diseñada por los desarrolladores y se observa de la siguiente manera.



Figura 11. Interfaz gráfica para App.

De las dos figuras anteriores se puede observar que, al presionar el botón 1 (Modo seguir) se envía a través de bluetooth el valor 1, el cual es leído por el puerto serial de arduino, este valor corresponde al valor que tomará la variable estado en el programa cuya función ya se mencionó.

III-B. Limitaciones y recomendaciones.

Las variables que almacenan la distancia de los ultrasónicos y los valores del modo a ejecutar, deben leerse durante cada condicional en el programa, es decir, deben estar leyendo en cada caso if, else if, etc, pues en caso de cambio se detecta inmediatamente y da la condición de salida al modo presente para así ingresar a otro modo.

Se debe tomar muy en cuenta a la hora de subir la programación al arduino, desconectar las señales de TX y RX del arduino con el módulo bluetooth Stereo, ya que estas se encuentran intercambiadas con respecto a las del módulo bluetooth, provocando errores en la carga de la progra, hacer esto evitará tener errores de memoria insuficiente en el arduino.

IV. RESULTADOS

Los resultados del sistema seguidor de música se muestran en las siguientes figuras.



Figura 12. Inicio del sistema WALL-E.

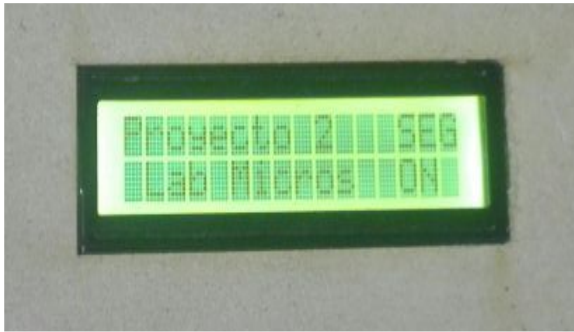


Figura 13. Modo seguir activado

Al activarse el modo seguir desde la aplicación, el sistema reproduce música solo en la habitación donde se detecta la persona, es decir, sigue la persona donde se desplace, este modo es perfecto cuando solo un individuo está en la casa.

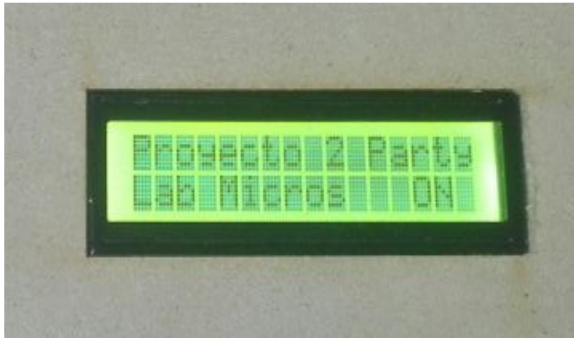


Figura 14. Modo Fiesta activado

Al activarse el modo fiesta, la música se reproducirá donde se detecte individuos, puede funcionar en varios inclusive, este modo es ideal cuando hay más de una persona en la casa, principalmente fiestas y eventos.



Figura 15. Modo Home activado

Al activarse este modo la música se reproducirá en toda la casa sin importar si hay o no personas en alguna área de la casa, reproduce música en todos los parlantes instalados.

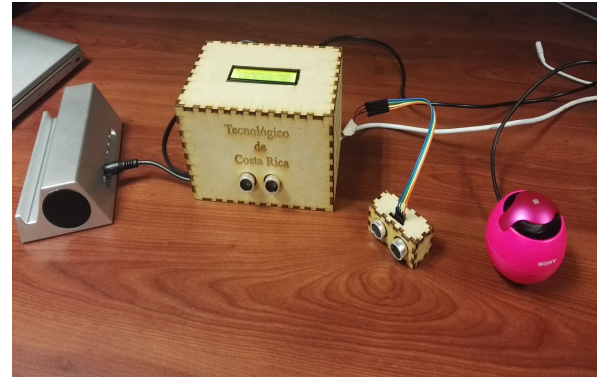


Figura 16. Prototipo Sistema de música Wall-E con dos parlantes

V. CONCLUSIONES

Se lograron cumplir los objetivos del proyecto, al implementar y diseñar el Sistema de música Wall-e satisfactoriamente. Subdividir el trabajo en etapas y luego unir las partes del trabajo en una sola, ayudó de manera muy significativa a la realización del proyecto evitando fuentes de error al ir probando cada etapa por separado.

Se puso en práctica los conocimientos con base a los microcontroladores, se reconocieron los pro y contras de trabajar en programación a alto nivel y su contraparte a bajo nivel para proyectos y sistemas con más capacidad y espacio de memoria, y necesidad de un sistema operativo propio, sin embargo para este proyecto fue muy conveniente trabajar con Arduino por su simplicidad y fácil manipulación al no requerir de tanta capacidad y memoria por el sistema de música Wall-E.

VI. REFERENCIAS

Acosta, A (2010). *Módulos Arduino: estructura típica. Soluciones Tecnológicas Aplicadas*.
Obtenido de
<http://www.noesys.com.ar/UTN/Embebidos-Civale/UD3.6-Modulos%20Arduino.pdf>

Cruz, J, (2012). *Simposio argentino de sistemas embebidos*. Universidad de Buenos Aires, Argentina. Obtenido de
http://www.sase.com.ar/2012/files/2012/09/Introduccion_a_los_Sistemas_Embebidos-SASE_2012.pdf

Simón, J. (2013). *Sobre la utilización de sistemas embebidos para la enseñanza de la programación en una carrera de Ingeniería Electrónica*. Universidad Nacional de Rosario.
Obtenido de
http://uea2013.frbb.utn.edu.ar/wp-content/uploads/S6_4.pdf

Referencia de Datasheet

Arduino Uno
<https://store.arduino.cc/usa/arduino-uno-rev3>
<http://datasheet.octopart.com/A000066-Arduino-datasheet-38879526.pdf>

Modulo Bluetooth Stereo XS3868
http://www.electrodragon.com/w/images/4/4b/BLK-MD-SPK-B_AT_Commands_OVC3860_List.pdf

Relay 4 Canales
<http://www.mantech.co.za/datasheets/products/COM-00154.pdf>

Modulo Bluetooth HC-05
<https://www.gme.cz/data/attachments/dsh.772-148.1.pdf>

Sensor Ultrasónico HC SR04
<http://www.electroschematics.com/8902/hc-sr04-datasheet/>