

Universidad Tecnológica Nacional

Proyecto Final

Control remoto IR vía WiFi

Autores:

- Luis Rubén Lebus

Director:

*Proyecto final presentado para cumplimentar los requisitos académicos
para acceder al título de Ingeniero _____*

en la

Facultad Regional Paraná

Julio de 2017

Declaración de autoría:

Yo/nosotros declaro/declaramos que el Proyecto Final “Control Remoto WiFi” y el trabajo realizado son propio/s. Declaro/declaramos:

- Este trabajo fue realizado en su totalidad, o principalmente, para acceder al título de grado de Ingeniero _____, en la Universidad Tecnológica Nacional, Regional Paraná.
- Se establece claramente que el desarrollo realizado y el informe que lo acompaña no han sido previamente utilizados para acceder a otro título de grado o pre-grado.
- Siempre que se ha utilizado trabajo de otros autores, el mismo ha sido correctamente citado. El resto del trabajo es de autoría propia.
- Se ha indicado y agradecido correctamente a todos aquellos que han colaborado con el presente trabajo.
- Cuando el trabajo forma parte de un trabajo de mayores dimensiones donde han participado otras personas, se ha indicado claramente el alcance del trabajo realizado.

Firmas:

-

Fecha:

AGRADECIMIENTOS

En primer lugar doy gracias a Esequiel, quien fue la persona con la que compartí hogar durante mi estadía en Paraná y la primera que me ayudó en mis inicios en la actividad universitaria.

Un especial agradecimiento a Santiago con quien compartí invaluables momentos y experiencias.

A "Correntino" quien junto a Nico y los mellizos me brindaron un segundo hogar siempre que lo necesité.

Luis Lebus

Universidad Tecnológica Nacional

Abstract

Facultad Regional Paraná

Ingeniero en Electrónica/Civil/Electromecánica

Control Remoto WiFi

Luis Rubén Lebus

Abstract:

This document presents the design and implementation of a device that allows to control home appliances that have infrared communication, such as televisions, DVD players, signal decoders, etc. The user performs the control through a mobile application that communicates over a WiFi link.

Keywords:

Communication, signal processing, infrared, protocol

Resumen:

En este documento se presenta el diseño e implementación de un dispositivo que permite controlar artefactos o electrodomésticos que poseen comunicación infrarroja, tales como televisores, reproductores de DVD, decodificadores de señal, etc. Para ello el usuario realiza el control mediante una aplicación móvil que se comunica con un dispositivo a través un enlace WiFi.

Palabras Clave:

Comunicación, procesamiento de señales, infrarrojo, protocolo

RECONOCIMIENTOS

A mis compañeros de facultad y posteriormente grandes amigos, sin ellos no hubiese sido posible mantenerme firme durante tantos años, en una carrera tan larga y con muchos momentos difíciles.

Luis Lebus

Índice

	<i>Página</i>
Capítulo 1: Introducción.....	16
Capítulo 2: Desarrollo.....	20
2.1 Descripción general.....	20
2.2 Unidad de procesamiento.....	20
2.3 Comunicación infrarroja.....	29
2.3.1 Señales infrarrojas.....	29
2.3.2 Modulación.....	29
2.3.3 Emisión.....	30
2.3.4 Recepción.....	30
2.3.5 Ensayos de controles remotos.....	32
2.3.6 Decodificación.....	39
2.3.7 Codificación.....	40
2.4 Comunicación WiFi.....	41
2.4.1 Protocolo de comunicación propio.....	43
2.4.2 Protocolo de transferencia de hipertexto.....	44
2.5 Aplicación móvil.....	45
2.6 Servidores.....	49
2.6.1 Servidor JSON.....	50
2.6.2 Servidor de imágenes.....	52
2.7 Control de cargas AC.....	53
2.8 Fuente de alimentación.....	54
2.9 Diseño completo.....	55
2.9.1 Hardware.....	55
2.9.2 Software.....	61
Capítulo 3: Resultados.....	63
Capítulo 4: Análisis de costo.....	64
Capítulo 5: Discusión y conclusión.....	66
Capítulo 6: Literatura citada.....	68

Lista de Figuras

Página

Figura N° 1: Captura 1 IR Remote Control	18
Figura N° 2: Captura 2 IR Remote Control	18
Figura N° 3: Captura 1 ASmart Remote IR	18
Figura N° 4: Captura 2 ASmart Remote IR	18
Figura N° 5: Captura 1 IR Universal TV Remote	19
Figura N° 6: Captura 2 IR Universal TV Remote	19
Figura N° 7: Esquema general del sistema completo.....	21
Figura N° 8: Microcontrolador PIC18F14K50.....	22
Figura N° 9: Placa 3D PIC18F14K50 (bottom).....	22
Figura N° 10: Placa 3D PIC18F14K50 (top).....	22
Figura N° 11: Placa real PIC18F14K50 (bottom).....	22
Figura N° 12: Placa real PIC18F14K50 (top).....	22
Figura N° 13: Esquemático placa PIC18F14K50.....	23
Figura N° 14: Placa Stellaris LunchPad.....	25
Figura N° 15: Placa 3D adaptadora (bottom).....	25
Figura N° 16: Placa 3D adaptadora (top).....	25
Figura N° 17: Placa real adaptadora (bottom).....	25
Figura N° 18: Placa real adaptadora (top).....	25
Figura N° 19: Esquemático placa adaptadora.....	26
Figura N° 20: Placa real adaptadora acoplada (bottom).....	27
Figura N° 21: Placa real adaptadora acoplada (top).....	27
Figura N° 22: Placa 3D adaptadora 2 (bottom).....	27
Figura N° 23: Placa 3D adaptadora 2 (top).....	27
Figura N° 24: Placa real adaptadora 2 (bottom).....	27
Figura N° 25: Placa real adaptadora 2 (top).....	27
Figura N° 26: Esquemático placa adaptadora 2.....	28
Figura N° 27: Esquema emisor y receptor infrarrojo.....	29
Figura N° 28: Circuito emisor de señal infrarroja.....	31
Figura N° 29: Esquema módulo receptor.....	31
Figura N° 30: Receptor IR 1.....	32
Figura N° 31: Receptor IR 2.....	32
Figura N° 32: Control remoto Reproductor DVD Tonomac.....	32
Figura N° 33: Señal control remoto Reproductor DVD Tonomac.....	33
Figura N° 34: Control remoto Decodificador RTC.....	33
Figura N° 35: Señal control remoto Decodificador RTC.....	33
Figura N° 36: Control remoto LCD Smart TV RCA.....	34
Figura N° 37: Señal control remoto LCD Smart TV RCA.....	34
Figura N° 38: Control remoto LCD BGH.....	34
Figura N° 39: Señal control remoto LCD BGH.....	35
Figura N° 40: Control remoto LCD Noblex.....	35
Figura N° 41: Señal control remoto LCD Noblex.....	35

Figura N° 42: Control remoto Reprod. VHS Philco.....	36
Figura N° 43: Señal control remoto Reproductor VHS Philco.....	36
Figura N° 44: Control remoto LCD Tonomac.....	36
Figura N° 45: Señal control remoto LCD Tonomac.....	37
Figura N° 46: Control remoto TV Tonomac Tubo.....	37
Figura N° 47: Señal control remoto TV Tonomac Tubo.....	37
Figura N° 48: Control remoto TV Philco Tubo.....	38
Figura N° 49: Señal control remoto TV Philco Tubo.....	38
Figura N° 50: Control remoto TV Samsung Tubo.....	38
Figura N° 51: Señal control remoto TV Samsung Tubo.....	39
Figura N° 52: Módulo ESP826.....	42
Figura N° 53: Distribución de pines ESP8266.....	42
Figura N° 54: Comandos AT de módulo ESP8266.....	42
Figura N° 55: Módulo RN171.....	43
Figura N° 56: Pantalla inicial de la aplicación.....	46
Figura N° 57: Pantalla de control de cargas AC.....	47
Figura N° 58: Pantalla de control aire acondicionado.....	47
Figura N° 59: Pantalla de “modo editor”.....	47
Figura N° 60: Pantalla de selección de ambiente.....	47
Figura N° 61: Pantalla de carga de un botón.....	48
Figura N° 62: Pantalla de ajustes.....	48
Figura N° 63: Pantalla menú contextual.....	49
Figura N° 64: Pantalla del control completo.....	49
Figura N° 65: Esquema de un servidor.....	50
Figura N° 66: Formato JSON.....	51
Figura N° 67: Pantalla descripción de modelo de un dispositivo.....	53
Figura N° 68: Circuito control AC.....	53
Figura N° 69: Implementación control AC (Bottom).....	54
Figura N° 70: Implementación control AC (Top).....	54
Figura N° 71: Batería BC50 Motorola.....	54
Figura N° 72: Circuito cargador de baterías.....	55
Figura N° 73: Placa 3D cargador de baterías.....	55
Figura N° 74: Placa real cargador de baterías.....	55
Figura N° 75: Placa definitiva 3D (bottom).....	56
Figura N° 76: Placa definitiva 3D (top).....	56
Figura N° 77: Placa definitiva real (bottom).....	57
Figura N° 78: Placa definitiva real (top).....	57
Figura N° 79: Placa definitiva real montada (bottom).....	57
Figura N° 80: Placa definitiva real montada (top).....	57
Figura N° 81: Placa definitiva acoplada a Stellaris (der).....	57
Figura N° 82: Placa definitiva acoplada a Stellaris (izq).....	57
Figura N° 83: Esquemático placa definitiva (Página 1).....	59
Figura N° 84: Esquemático placa definitiva (Página 2).....	60
Figura N° 85: Diagrama de flujo de firmware (1).....	61
Figura N° 86: Diagrama de flujo de firmware (2).....	62

Lista de Tablas

Página

Tabla N° 1: Descripción de pines ESP8266.....	43
Tabla N° 2: Costos de componentes del proyecto.....	64
Tabla N° 3: Costos de componentes del proyecto reducido.....	65

Lista de Abreviaciones

IR: Infrarrojo

IdC: Internet de las cosas

AC: Corriente alterna

RAM: Memoria de acceso aleatorio

ROM: Memoria de acceso único

EEPROM: Memoria de acceso único eléctricamente borrable

SMD: Dispositivo de montaje superficial

PWM: Modulación de ancho de pulso

AGC: Control automático de ganancia

HTTP: Protocolo de transferencia de hipertexto

MAC: Control de acceso al medio

XML: Lenguaje de marcado extensible

JSON: Notación objeto JavaScript

RMS: Valor cuadrático medio

DEDICATORIA

A mi madre Nidia quien ha sido un pilar fundamental en mi formación como profesional, por brindarme la confianza, consejos y recursos para lograrlo.

A mi padre Edicto, la persona a quien más admiro, de la cual he aprendido que la perseverancia es el camino asegurado al éxito y que espero, algún día, llegar a ser al menos la mitad de lo que él representa.

A mi hermana Susana, que siempre estuvo acompañándome en mi trayecto universitario, desde aquel día en el que buscábamos mi primer departamento.

Luis Lebus

Capítulo 1: Introducción

El ser humano siempre ha estado y estará imaginando cómo será el futuro, es lo que le permite seguir evolucionando, creando y también, condenándose a un futuro que siempre está llegando. En este aspecto, la domótica es ese campo que más está aprovechando dichas ideas de futuro aunque también uno de los que más sufre la velocidad de innovación y la falta de un consenso o estandarización de las normas que se crean.

Años atrás ya se soñaba con hogares automatizados, futuristas, capaces de ofrecer las mayores comodidades a sus habitantes. Y uno de los puntos que más visiones generaba era la cocina. Muebles que tras insertar una tarjeta preparaban la receta, cocinas de inducción para calentar los alimentos y mecanismos con los que la comida llegaba sola a la mesa. La cocina siempre ha sido uno de los puntos del hogar más usados para mostrar el futuro. Ideas que en aquel momento dejarían sorprendidas a muchas amas de casa pero que con el tiempo se han ido desdibujando. Se ha sobrepasado el año 2000 y aún no se ha llegado a esas ideas futuristas, sin embargo lo que quedó en bocetos dio paso a otras ideas que sobrepasan lo imaginado.

Si se observa la actualidad de las últimas ferias de electrónica de consumo, como IFA, seguro que se reconocerán mucho de los dispositivos de los que se hace mención más arriba. En estos momentos es posible encontrar una gran variedad de aparatos capaces de ser controlados a través de aplicaciones para teléfonos inteligentes. Existen bombillas, interruptores, persianas, cerraduras e incluso cafeteras entre otros tantos *gadgets*.

Mediante el uso de la conectividad WiFi o Bluetooth, estos dispositivos permiten ser controlados remotamente desde una aplicación instalada en nuestro terminal móvil. Por ejemplo, se puede configurar el color y potencia de las bombillas Philips HUE, modificar la temperatura de un aire acondicionado a través de un termostato como NEST o recibir una notificación una vez finalice el lavado en la nueva Samsung WW9000. Esta forma de automatizar y gestionar datos se engloba en un concepto muy moderno denominado “internet de las cosas”, el cual hace referencia a la forma en que objetos cotidianos se conectan con internet y entre ellos (Santamaría, 2014).

Se estima que Internet de las cosas (IdC), algunas veces denominado "Internet de los objetos", lo cambiará todo, incluso a los comportamientos humanos. Si bien puede parecer una declaración arriesgada, hay que tener en cuenta el impacto que Internet ha tenido sobre la educación, la comunicación, las empresas, la ciencia, el gobierno y la humanidad. Claramente Internet es una de las creaciones más importantes y poderosas de toda la historia de la humanidad.

Ahora, se debe tener en cuenta que IdC representa la próxima evolución de Internet, que será un enorme salto en su capacidad para reunir, analizar y distribuir datos que pueden ser convertidos en información, conocimiento y en última instancia, sabiduría. En este contexto, IdC se vuelve inmensamente importante.

Ya están en marcha proyectos de IdC que prometen cerrar la brecha entre ricos y pobres, mejorar la distribución de los recursos del mundo para quienes más los necesitan y ayudarnos a comprender el planeta para que podamos ser más proactivos y menos reactivos. Aun así, son varias las barreras que amenazan con retrasar el desarrollo de IdC, como la transición a IPv6, el establecimiento de un conjunto de normas en común y el desarrollo de fuentes de energía para millones de sensores diminutos (Evans, 2011).

Siguiendo las tendencias y el concepto de IdC, en este proyecto se diseñó e implementó un dispositivo que permite controlar artefactos o electrodomésticos que poseen comunicación infrarroja (IR), tales como, televisores, reproductores de DVD, decodificadores de señal, etc. El usuario realiza el control mediante una aplicación móvil que se comunica con un dispositivo a desarrollar, a través un enlace WiFi.

Se puede deducir, fácilmente, que este sistema está orientado a mejorar el confort en el hogar. En la actualidad es muy común que las personas posean un teléfono móvil inteligente, de modo que sería muy útil utilizar este teléfono para reemplazar los controles remotos IR convencionales, más aún cuando en una misma habitación se tienen varios artefacto IR, como puede ser un televisor, un reproductor de DVD, un reproductor de música y un aire acondicionado. El usuario puede personalizar la aplicación móvil cargando solamente los botones que suele utilizar de cada control remoto. De este modo, en una sola pantalla del móvil puede comandar todos sus artefactos. Además, se le agrega la posibilidad de controlar cargas de corriente alterna (AC) tales como lámparas, ventiladores, etc.

En la actualidad existen una gran cantidad de aplicaciones móviles que permiten controlar dispositivos del hogar como televisores, aires acondicionados, etc. Como ejemplo, se mencionan a continuación los tres primeros resultados luego de realizar una búsqueda en Play Store:

IR Remote Control

Esta aplicación permite utilizar el teléfono como un control remoto para TV, DVD, VCR. La misma utiliza el emisor infrarrojo presente en el móvil y actualmente pueden ser controlados los siguientes fabricantes de dispositivos.

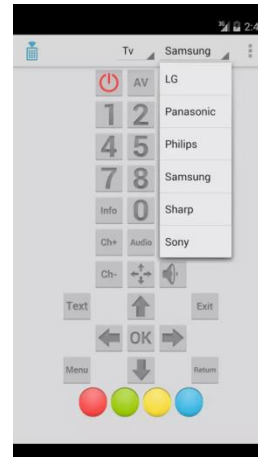
- LG
- Panasonic
- Samsung
- Sony

Para ser capaz de utilizar esta aplicación se debe contar con un dispositivo móvil con sistema operativo Android 4.4 (KitKat) o superior con transmisor infrarrojo incluido. Esta aplicación es ofrecida por Johan Westling y posee una calificación de 3,2/5. En las figuras N° 1 y N° 2 se puede observar la apariencia y como se selecciona un dispositivo deseado.

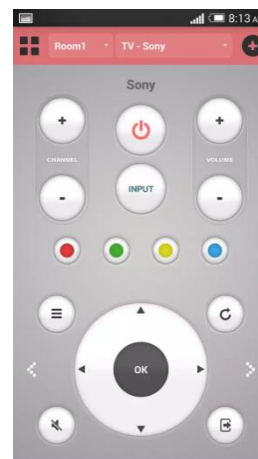
ASmart Remote IR

Esta aplicación es ofrecida por NXRsoft y posee una calificación 3,7/5. Esta aplicación sólo funciona con sistema Android que posean transmisor infrarrojo incorporado. Los dispositivos que pueden ser controlados son, por ejemplo:

- Televisores.
- Decodificadores de señal.
- Proyector.
- Reproductores de DVD.

Figura N° 1: Captura 1 IR Remote Control.^[11]Figura N° 2: Captura 2 IR Remote Control.^[11]

Esta aplicación tiene la característica de poder definir “habitaciones” o “ambientes” y configurar los dispositivos existentes en cada lugar. En las figuras N° 3 y N° 4 se puede observar cómo se presenta la aplicación.

Figura N° 3: Captura 1 ASmart Remote IR.^[12]Figura N° 4: Captura 2 ASmart Remote IR.^[12]

IR Universal TV Remote

Esta aplicación busca reemplazar los controles remotos físicos de televisores, además de otros dispositivos como acondicionadores de aire, ventiladores, luces, entre otros. El funcionamiento se basa en la utilización de un transmisor infrarrojo presente en los dispositivos móviles como teléfonos y tablets.

Esta aplicación es altamente personalizable, por ejemplo, se puede cambiar el color de la barra de herramientas, modificar el color de fondo o agregar una imagen, se puede modificar los códigos IR, colores, textos y posición de los botones. Se puede programar para se envíen múltiples comandos con un único clic de un botón.

Esta aplicación está ofrecida por WaveSpark y posee una calificación de 3,7/5. En las figuras N° 5 y N° 6 se observa la misma.

Figura N° 5: Captura 1 IR Universal TV Remote.^[13]Figura N° 6: Captura 2 IR Universal TV Remote.^[13]

Tanto las tres aplicaciones mencionadas, como la mayoría que se puede encontrar en la tienda de Android, poseen una o más de las tres características que se nombran a continuación:

1. Es necesario que el teléfono móvil o *tablet* tenga un emisor infrarrojo.
2. Se basan, exclusivamente, en bases de datos para obtener la información de las señales de los botones de un control remoto. En muchos casos, estas bases de datos se encuentran almacenadas dentro de la misma aplicación.
3. La aplicación es poco personalizable, esto es, se tiene una pantalla por electrodoméstico con botones ya definidos.

Teniendo en cuenta lo mencionado anterior, se describen a continuación las características fundamentales que tendrá este sistema, que en cierta forma busca solventar los inconvenientes y diferenciarse de las demás aplicaciones móvil.

1. Este sistema contará con un dispositivo que se conectará a la red WiFi del hogar, el mismo se comunicará con una aplicación móvil que correrá en teléfonos o *tablets*. Este dispositivo será el encargado de emitir y recibir señales infrarrojas, por lo tanto ya no es necesario que el móvil cuente con un transmisor incorporado.
2. De forma similar, este sistema obtendrá la información para emitir las señales infrarrojas a partir de una base de datos. Sin embargo, esta no se ubicará en la aplicación o en el móvil sino que se almacenará en un servidor en Internet desde donde la aplicación descargará solamente los datos de los botones que son utilizados. De esta forma se evita tener almacenado en el teléfono datos que nunca se utilizaran.

Por otro lado, esta base de datos se irá actualizando por los mismos usuarios ya que el sistema contará con un receptor infrarrojo que decodificará la señal de un botón de un control remoto dado y almacenará los datos en el servidor para que otros usuarios puedan utilizarlos. De esta forma, a medida que más usuarios utilizan este sistema, la base de datos crecerá más rápidamente.

3. La aplicación que se desarrollará tendrá como característica que el usuario podrá seleccionar cuantos botones quiere en pantalla, la ubicación, el texto y función de manera que desde una sola pantalla podrá comandar distintos artefactos.

Capítulo 2: Desarrollo

2.1 Descripción general

El sistema se basa en el desarrollo de dos partes principales, a continuación se mencionan en forma resumida.

- **Aplicación móvil:** Se desarrolló una aplicación para teléfonos móviles para sistema operativo Android, la cual posee botones y una apariencia similar a la de un control remoto IR convencional. La aplicación se comunica con un dispositivo que funciona como interfaz IR-WiFi.
- **Interfaz IR-WiFi:** Este dispositivo tiene la función de recibir y decodificar señales infrarrojas provenientes de controles remotos convencionales, las almacena y puede emitir las mismas cuando la aplicación móvil lo indique. Este dispositivo está basado en la placa de desarrollo Stellaris Launchpad de Texas Instruments a la cual se le incorpora un receptor y emisor IR, un módulo WiFi, entre otros componentes.

La figura N° 7 muestra un esquema general del sistema completo. En las siguientes páginas se describirán los bloques que conforman el sistema.

2.2 Unidad de procesamiento

Como elemento de procesamiento principal se utilizó un microcontrolador, al cual se le asignó las siguientes funciones principales:

- Decodificar las señales provenientes del receptor IR.
- Codificar y generar la trama IR para enviar al emisor.
- Interactuar con la aplicación móvil mediante el enlace WiFi.
- Almacenar y recuperar la información sobre los botones utilizados.

En una primera instancia se decidió utilizar un microcontrolador PIC18F14K50, de la Compañía Microchip. La elección del mismo se debió a la facilidad de programación, la documentación existente, el bajo costo y la posibilidad de adquirirlo fácilmente en el país. La figura N° 8 muestra la distribución de los pines del mismo.

Entre las características de este microcontrolador se encuentran las siguientes:

- 16 KB de memoria de programa.
- 1 EUSART, 1 SPI, 1 I2C.
- 1 timer de 8 bits y 3 timers de 16 bits.
- 11 canales de ADC de 10 bits y 2 comparadores analógicos.
- USB.
- Capacidad de trabajar a 48 MHz con PLL interno.
- Bajo consumo.

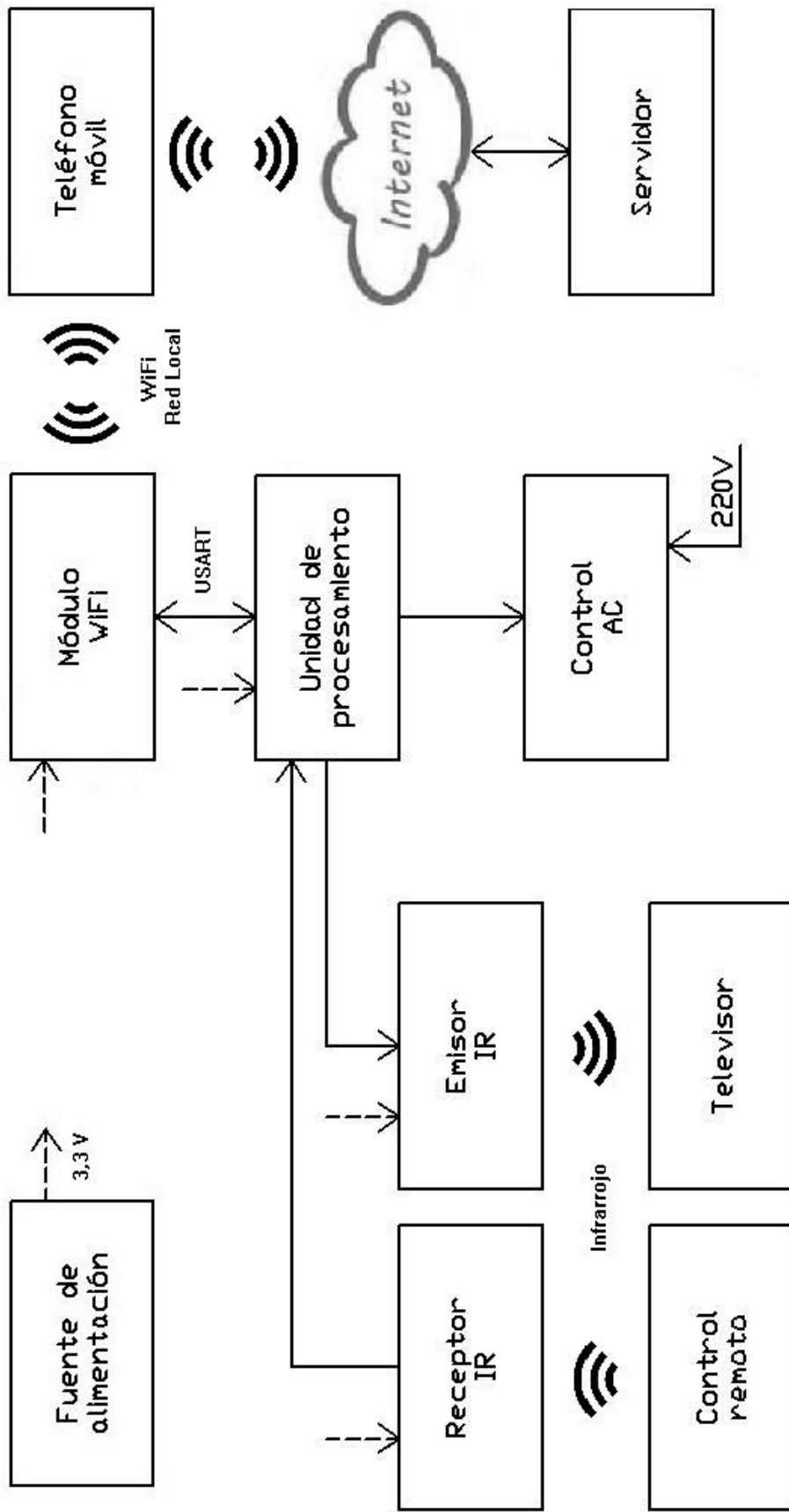


Figura N° 7: Esquema general del sistema completo.

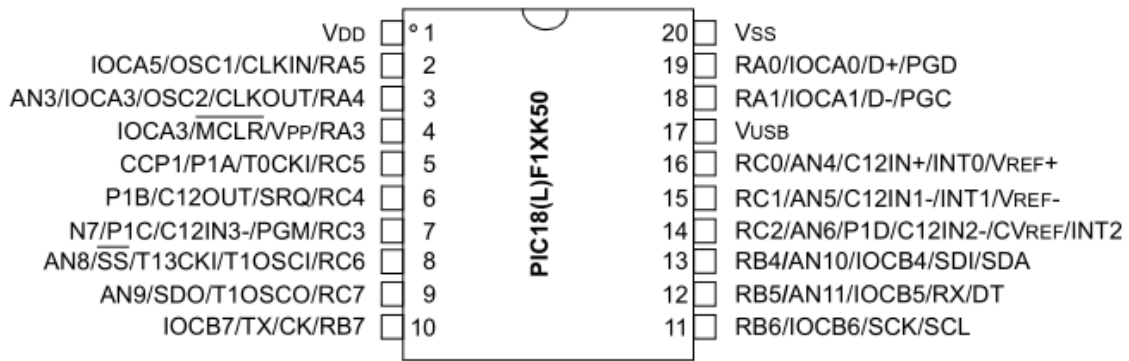


Figura N° 8: Microcontrolador PIC18F14K50.^[14]

Para poder realizar los primeros ensayos se diseñó y fabricó una pequeña placa de desarrollo la cual incluía un receptor infrarrojo, el circuito emisor, el microcontrolador mencionado anteriormente y un conversor RS232/TTL, este último componente es necesario para comunicarse con una computadora mediante el puerto serial y realizar el *debug* por consola. En las figuras N° 9 y N° 10 puede observarse la visualización 3D, mientras que en las figuras N° 11 y N° 12 se muestra la placa fabricada mediante el método casero clásico. Por otro lado, la figura N° 13 muestra el esquemático de la placa mencionada. El software utilizado en el diseño de todas las placas en este proyecto fue EAGLE Professional Edition, mientras que para generar la representación 3D se utilizó el software SketchUP de Google. Los circuitos y componentes utilizados se describirán en las páginas siguientes.

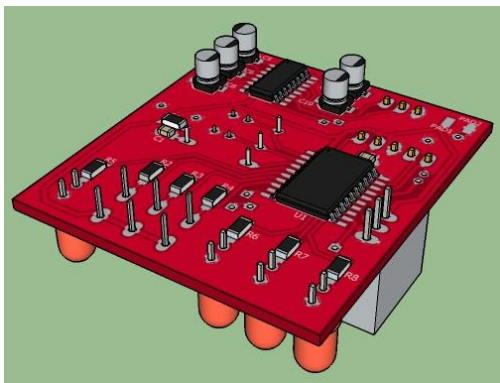


Figura N° 9: Placa 3D PIC18F14K50 (bottom).

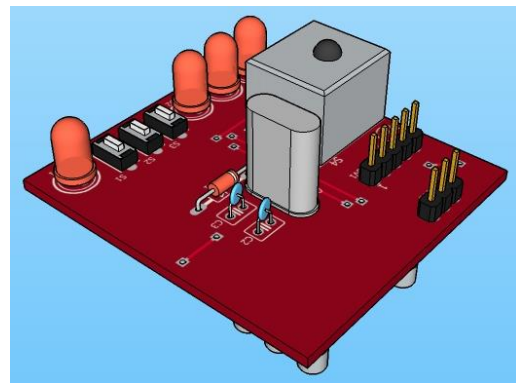


Figura N° 10: Placa 3D PIC18F14K50 (top).

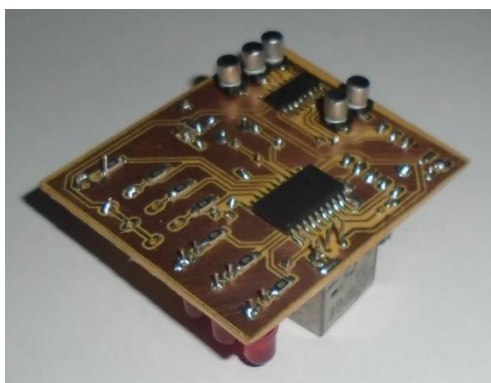


Figura N° 11: Placa real PIC18F14K50 (bottom).



Figura N° 12: Placa real PIC18F14K50 (top).

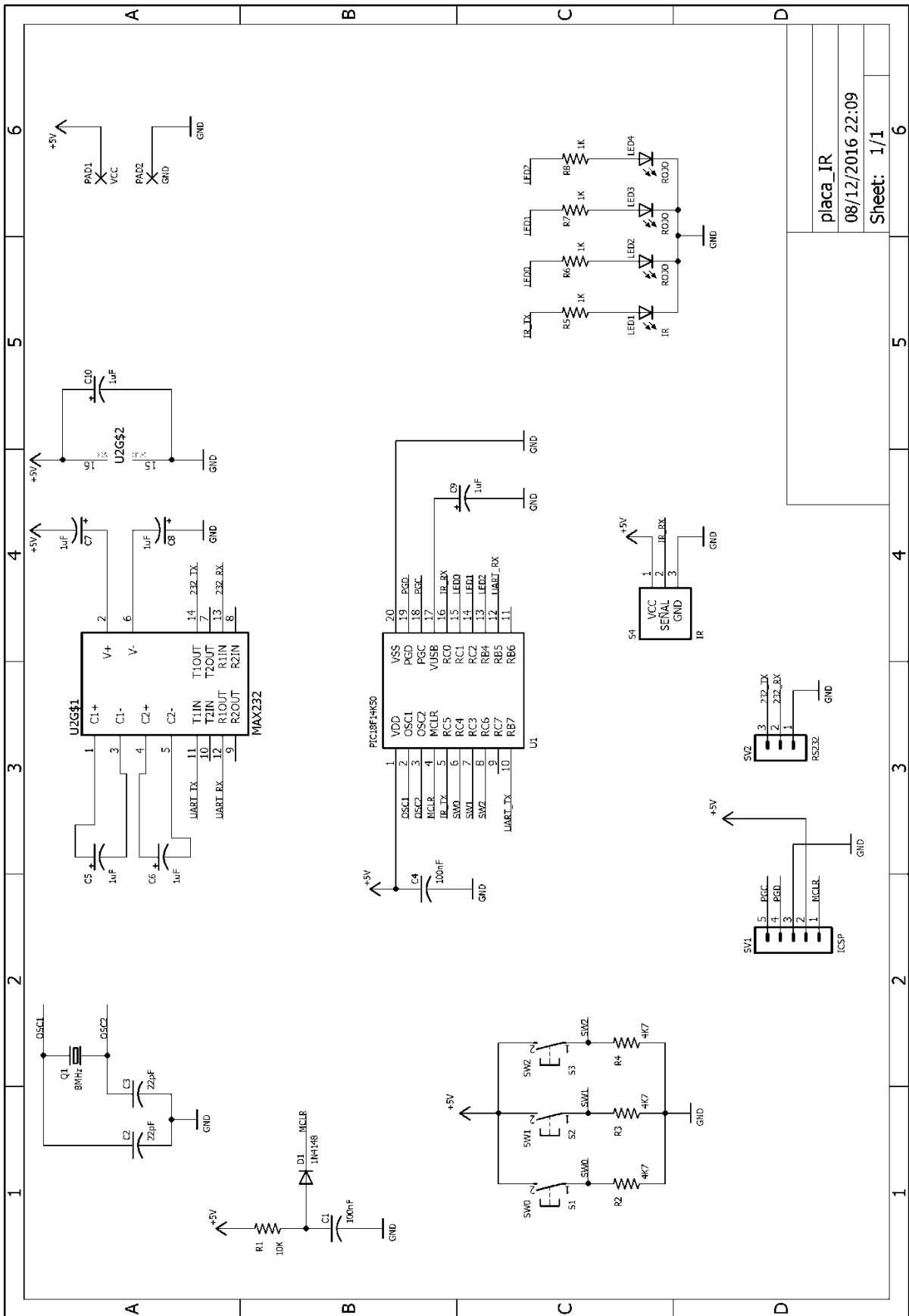


Figura N° 13: Esquemático placa PIC18F14K50.

placa_IR
08/12/2016 22:09
Sheet: 1/1

Con la placa mencionada se procedió a realizar ensayos de recepción de señales IR, capturando los tiempos y forma de la señal, y visualizando los resultados por consola. Para ello se utilizó el software Cool Term.

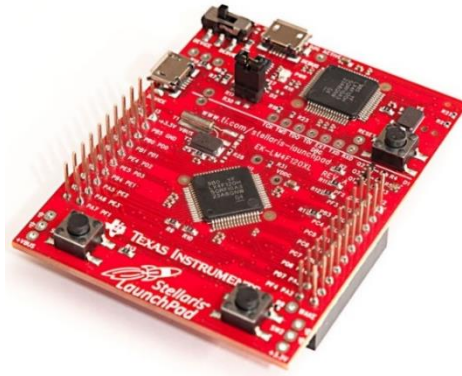
Si bien se logró que la decodificación sea exitosa, contrastada con las mediciones realizadas mediante osciloscopio, se presentaron importantes dificultades al momento de corregir errores ya que este microcontrolador no permite realizar el *debug* mediante el entorno de desarrollo, el único método es utilizar el módulo EUSART para mostrar mensajes por consola. Esto motivó a la elección de otro microcontrolador.

Para superar el inconveniente anterior, se decidió utilizar la plataforma Stellaris LunchPad, la misma es una placa de desarrollo de bajo costo desarrollada por la compañía Texas Instruments. Esta cuenta con el microcontrolador LM4F120H5QR. La figura N° 14 muestra la placa nombrada.

A continuación se mencionan las características más relevantes.

- Microcontrolador con núcleo ARM® Cortex™-M4 con las siguientes características:
 - Operaciones de punto flotante
 - Velocidad de operación máxima de 80 MHz
 - Memoria flash de 256 KB
 - Memoria EEPROM de 2 KB
 - Memoria SRAM de 32 KB
 - 8 UARTs
 - 4 I2C
 - 4 SPI
 - 27 Temporizadores
- Interfaz ICDI (In-Circuit Debug Interface) USB incluida en la tarjeta.
- Conexión micro USB-B para depuración.
- Conexión micro USB-B de dispositivo para aplicaciones de usuario.
- Switch selector de alimentación de dispositivo/depuración.
- Pines BoosterPack XL, compatible con tarjetas BoosterPack existentes.
- 2 botones para aplicaciones de usuario.
- Botón de reset.
- 1 LED RGB.
- Puente (jumper) para medición de corriente.
- Cristal principal de 16 MHz.
- Cristal de 32.768 KHz para hibernación y reloj en tiempo real (RTC – Real Time Clock).

Para la realización de los ensayos se diseñó e implementó una placa adaptadora que contenga al emisor y receptor IR, el módulo WiFi y dos pulsadores. Estos últimos son utilizados para iniciar el módulo WiFi en modo “flash”, lo cual se describirá más adelante. La placa cuenta, además, con un

Figura N° 14: Placa Stellaris LunchPad.^[15]

regulador de 3,3[V], y se acopla al Stellaris LunchPad mediante los conectores de expansión. En las figuras N° 15 y N° 16 se muestra la representación 3D, mientras que en las figuras N° 17 y N° 18 se observa la placa real. Por otro lado, la figura N° 19 muestra el esquemático de la misma.

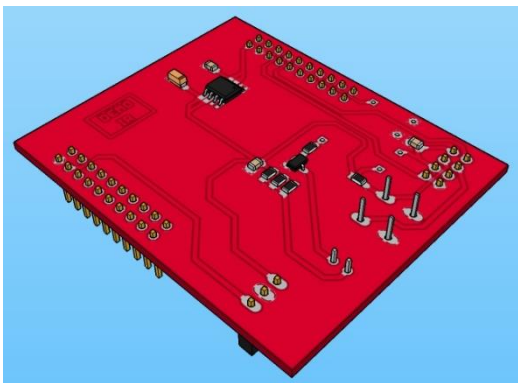


Figura N° 15: Placa 3D adaptadora (bottom).

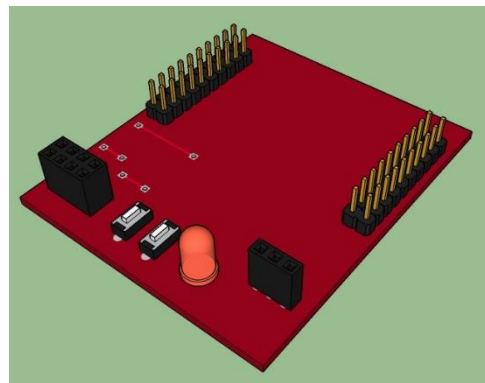


Figura N° 16: Placa 3D adaptadora (top).

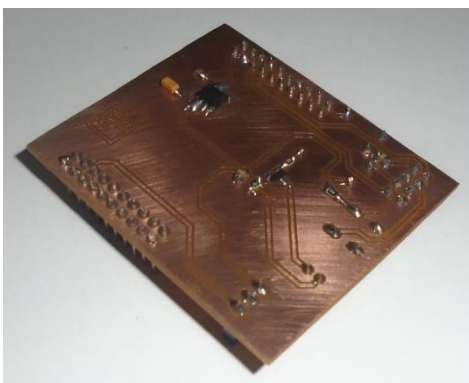


Figura N° 17: Placa real adaptadora (bottom).

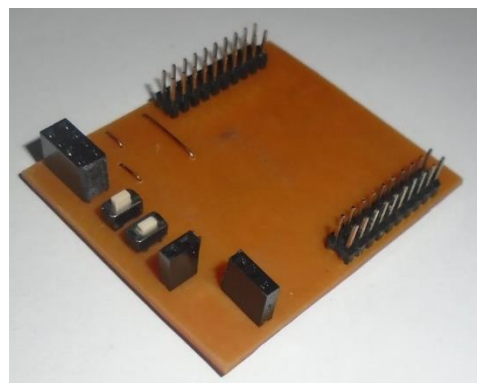
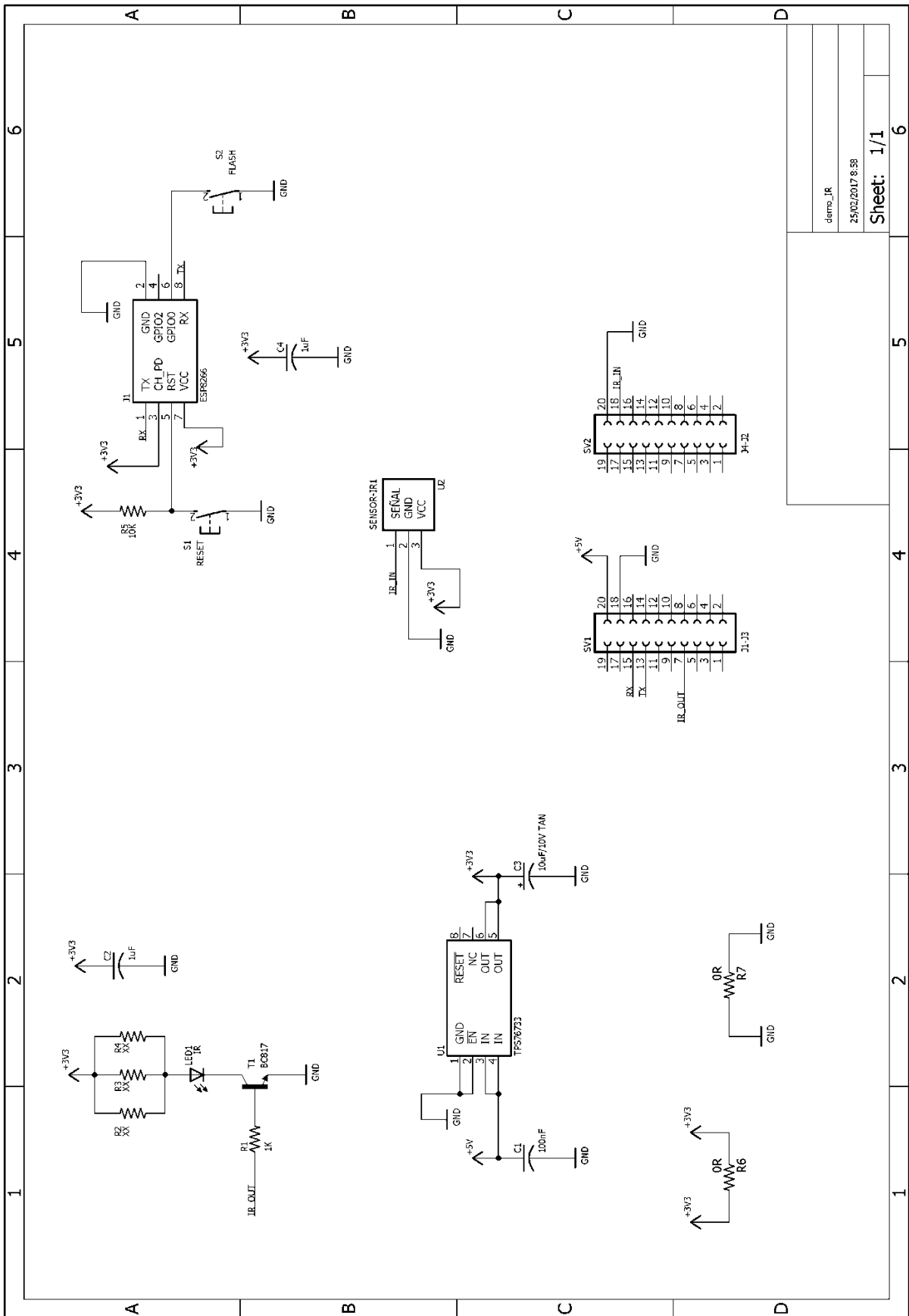


Figura N° 18: Placa real adaptadora (top).



Sheet: 1/1	
demo_IR	
25/02/2017 8:58	

Figura N° 19: Esquemático placa adaptadora.

En las figuras N° 20 y N° 21 se observa cómo se acopla a la placa de desarrollo Stellaris.

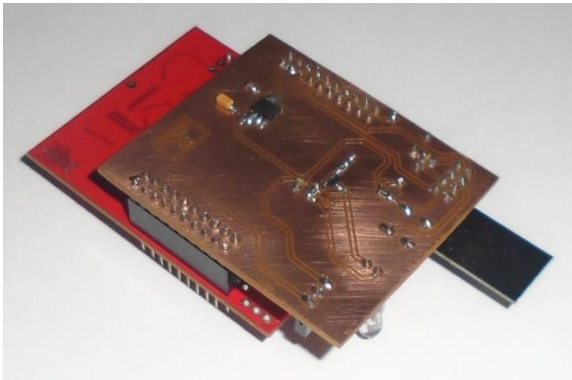


Figura N° 20: Placa real adaptadora acoplada (bottom).

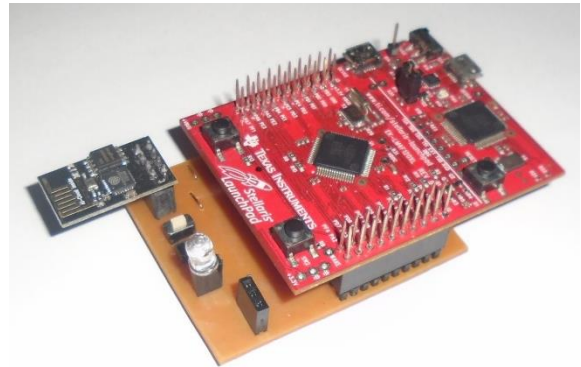


Figura N° 21: Placa real adaptadora acoplada (top).

Además de las placas mencionadas, se diseñó y fabricó una segunda placa adaptadora cuya representación 3D e implementación se observa en las figuras N° 22, N° 23, N° 24 y N° 25, mientras que la figura N° 26 se muestra el esquemático de la misma. Esta placa se utilizó junto a la anterior para comprobar el funcionamiento del dispositivo en múltiples ambientes del hogar en forma simultánea. Esto se desarrollará cuando se describa el diseño de la aplicación móvil.

Esta placa se diferencia de la anterior en que aquí se eliminaron los pulsadores y el receptor IR, mientras que se agregó un *buzzer* para indicar cuando el sistema recibe un comando y se utilizaron dos leds infrarrojos en vez de solamente uno.

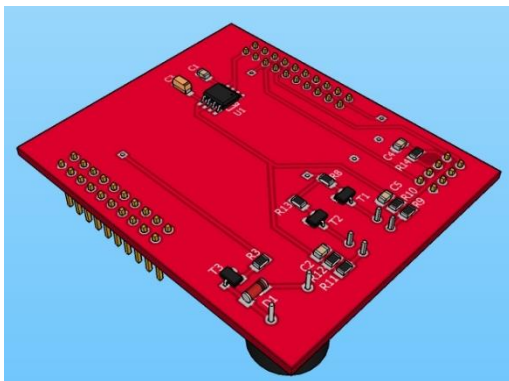


Figura N° 22: Placa 3D adaptadora 2 (bottom).

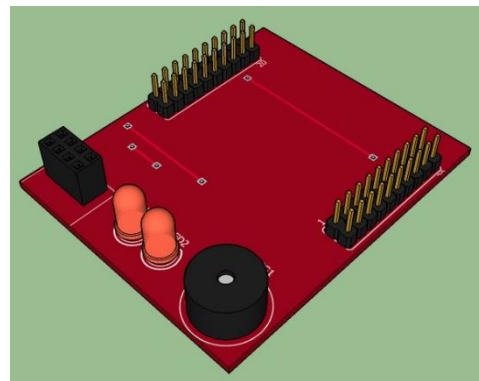


Figura N° 23: Placa 3D adaptadora 2 (top).

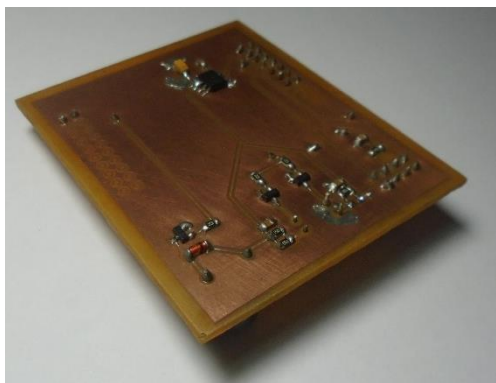


Figura N° 24: Placa real adaptadora 2 (bottom).

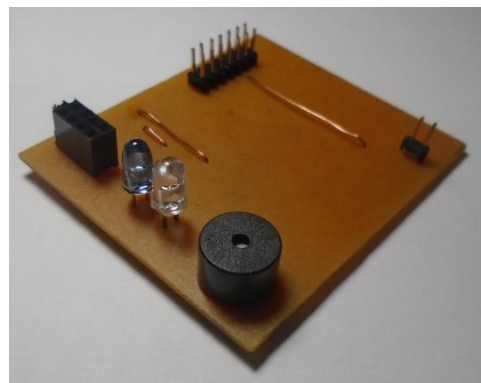
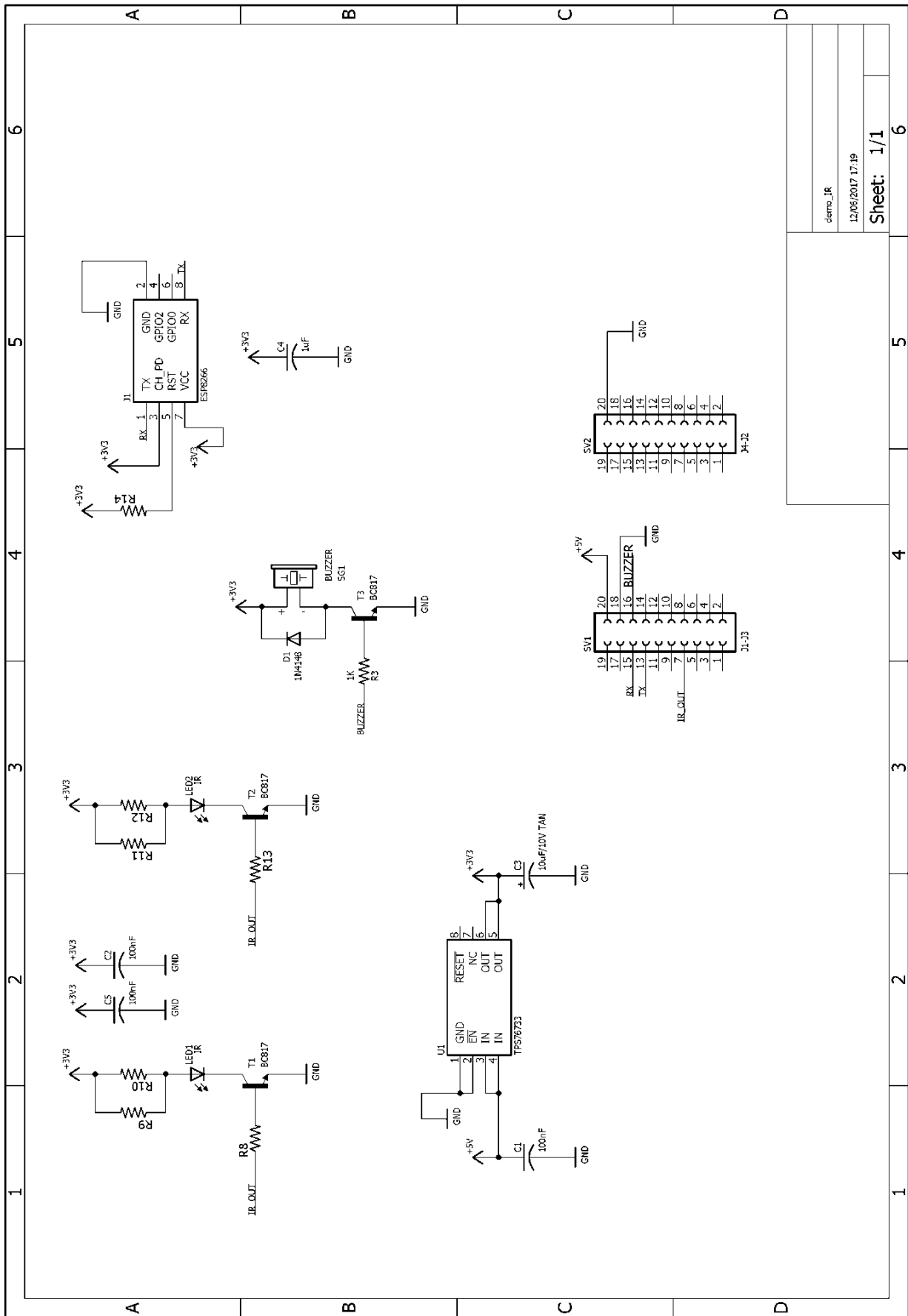


Figura N° 25: Placa real adaptadora 2 (top).



derro_IR	
12/06/2017 17:19	
Sheet: 1/1	6

Figura N° 26: Esquemático placa adaptadora 2.

Tal como se mencionó anteriormente, este microcontrolador posee una memoria EEPROM interna de 2KB. Se utilizó este recurso para almacenar la información de los últimos 30 botones emitidos, para esto se utiliza un buffer circular.

Cuando se energiza el dispositivo el sistema inicializa los periféricos, luego de ello se procede a leer por completo el bloque de EEPROM utilizado y se copia esa información a un vector en memoria RAM, esto incrementa la velocidad de respuesta al momento de emitir una señal, cuando la aplicación móvil lo solicita. Si el usuario presiona un botón en su teléfono y el microcontrolador no cuenta con la información para emitir la señal en su memoria, lo solicitará a la aplicación móvil y esta procederá a proporcionar los datos necesarios. A continuación, el microcontrolador almacena la información recibida en el siguiente lugar disponible dentro del bloque memoria EEPROM. Si se alcanza el límite de 30 lugares se comienza desde el principio.

2.3 Comunicación infrarroja

2.3.1 Señales infrarrojas

Las emisiones infrarrojas son especialmente adecuadas para el control remoto por ser invisibles, inocuas y fácilmente generables mediante leds IR. A esto se debe su presencia en prácticamente todos los dispositivos de audio, video y TV ofrecidos en el mercado. Precisamente, debido a la utilización masiva de estos dispositivos es necesario codificar de algún modo las señales para que el sistema sea inmune a disparos no deseados provocados por otros controles, así como a las emisiones infrarrojas de objetos luminosos y calientes en general.

2.3.2 Modulación

Deben distinguirse dos clases de codificación. En primer lugar la señal IR debe ser modulada, es decir que esta se enciende y apaga a una frecuencia determinada para poder diferenciar la señal válida de todas las emisiones infrarrojas existentes en el ambiente, las que generalmente serán continuas o de frecuencia aleatoria. Esta señal periódica se conoce como señal portadora y su frecuencia se suele ubicar entre 30[KHz] y 60[KHz]. Sin embargo, el valor más común es 38[KHz].

Además de generar la señal anterior, es necesario emitirla e interrumpirla en lapsos pre-establecidos de antemano para enviar o recibir comandos que el receptor pueda entender. Lo anterior queda determinado por el protocolo de comunicación utilizado (Grassi, s.f.).

En la figura N° 27 se puede ver una señal modulada que es emitida por el led IR del transmisor en el lado izquierdo. La señal detectada sale del receptor en el otro lado.

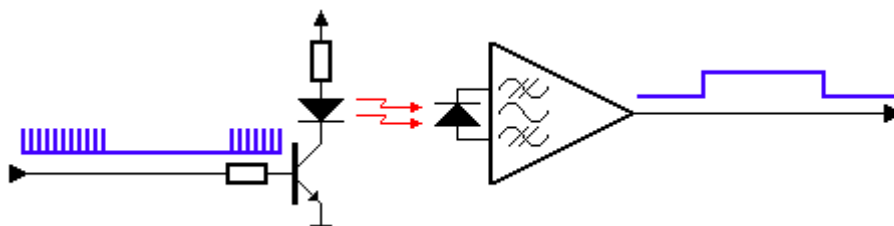


Figura N° 27: Esquema emisor y receptor infrarrojo.^[5]

2.3.3 Emisión

La forma más fácil y común de generar o transmitir una señal infrarroja es mediante un led infrarrojo. En líneas generales, se puede mencionar que los leds se usan como indicadores en muchos dispositivos y en iluminación. Los primeros ledes emitían luz roja de baja intensidad, no obstante los dispositivos actuales emiten una luz de alto brillo en el espectro infrarrojo, visible y ultravioleta.

Debido a su capacidad de operación a altas frecuencias, son también útiles en tecnologías avanzadas de comunicaciones y control. Los ledes infrarrojos también se usan en unidades de control remoto de muchos productos comerciales incluyendo equipos de audio y video, tal como se mencionó anteriormente (Wikipedia, 2017).

Normalmente, el transmisor es un dispositivo alimentado por medio de pilas o baterías, por lo que debe el consumo debe ser el menor posible. Por otro lado, la señal de IR también debe ser lo más intensa posible para lograr una distancia de control aceptable.

Muchos chips están diseñados para ser utilizados como transmisores de infrarrojos. Antiguamente, los chips estaban dedicados a sólo uno de los muchos protocolos que se inventaron. Hoy en día se utilizan microcontroladores de muy baja potencia en transmisores IR por la simple razón de que son más flexibles en su uso. Cuando no se presiona ningún botón, se encuentran en un modo de suspensión de muy baja potencia, en el que casi no se consume ninguna corriente. El procesador se despierta para transmitir el comando IR apropiado sólo cuando se pulsa una tecla.

La corriente a través del led puede variar de $0,1[A]$ hasta $1[A]$, con el fin de obtener una distancia de control aceptable. Estas corrientes pueden ser tan altas debido a que los impulsos que conduce el led son muy cortos. Es bastante común reducir el ciclo de trabajo de la señal portadora a $1/3$ o incluso $1/4$, esto reduce los requisitos de alimentación y se puede aumentar la corriente a través del led sin sobrecalentarlo para obtener una mayor distancia de control (Bergmans, 2002).

El circuito utilizado para emitir las señales infrarrojas es el que se observa en la figura N° 28. Las resistencias que regulan la corriente del led son de un valor de $15[\text{ohm}]$, este se definió a partir de sucesivos ensayos hasta lograr un comportamiento adecuado. Como se observa, se utilizaron dos resistencias en paralelo con el fin de reducir la corriente en las mismas ya que se usaron resistencias SMD 0805 de $1/10[W]$.

Además, se realizaron pruebas con la segunda placa adaptadora construida la cual poseía dos leds IR orientados en dirección opuesta entre sí, con un ángulo de 45° con respecto a la vertical. Esto fue realizado con la intención de mejorar la distribución de la emisión y así lograr que el emisor sea omnidireccional, sin embargo no se obtuvieron mejoras relevantes con respecto al uso de un sólo led.

2.3.4 Recepción

En la figura N° 29 se puede observar un típico diagrama de bloques de un receptor IR. La señal recibida es captada por el diodo de detección IR en el lado izquierdo del diagrama. Esta señal es amplificada y limitada en las primeras dos etapas. El limitador actúa como un control de ganancia

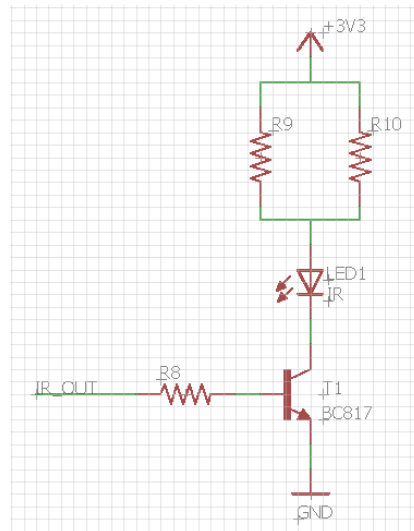


Figura N° 28: Circuito emisor de señal infrarroja.

automático (AGC) para obtener un nivel de pulso constante, independientemente de la distancia al teléfono.

Como se puede ver sólo la señal alterna se envía al filtro de pasa-banda, el cual se sintoniza a una frecuencia portadora determinada. Estas varían de 30[KHz] a 60[KHz] por lo que es importante elegir el valor correcto. Las siguientes etapas son un detector, un integrador y un comparador. El propósito de estos tres bloques es detectar la presencia de la frecuencia portadora, en donde si ésta se encuentra presente, el comparador establece un nivel bajo en su salida. Todos estos bloques están integrados en un solo componente electrónico.

Hay varios fabricantes de receptores IR en el mercado. Siemens, Vishay y Telefunken son los principales proveedores. Siemens tiene su serie SFH506-xx, donde xx denota la frecuencia portadora de 30, 33, 36, 38, 40 o 56kHz. Telefunken tiene sus series TFMS5xx0 y TK18xx, donde xx indica de nuevo la frecuencia de portadora a la que está ajustado el dispositivo. Por otro lado, Vishay posee dispositivos como TSOP12xx, TSOP48xx y TSOP62xx (Bergmans, 2002).

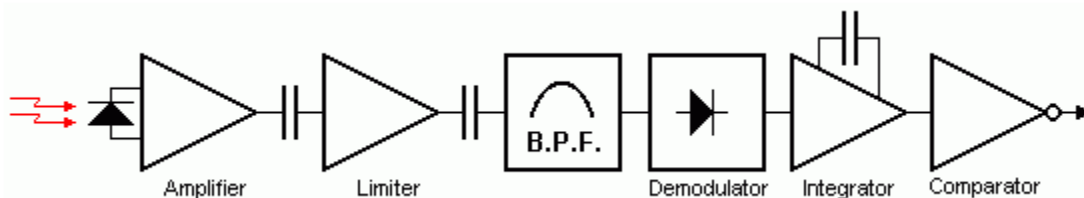


Figura N° 29: Esquema módulo receptor.^[5]

En este proyecto se utilizó en primer lugar el receptor mostrado en la figura N° 30, sin embargo se encontraron dos inconvenientes:

- No puede alimentarse con 3,3[V], lo cual lo hace incompatible con la placa Stellaris.
- En determinados tipos de señales, los pulsos tienden a deformarse en lugar de mantener la forma de señal cuadrada, lo que produce que la medición no sea fiable.

Debido a esto, se optó por utilizar el receptor mostrado en la figura N° 31 el cual no presentó las desventajas descriptas.



Figura N° 30: Receptor IR 1.



Figura N° 31: Receptor IR 2.

2.3.5 Ensayos de controles remotos.

A continuación se detallarán los resultados de ensayos que se hicieron a diversos controles remotos. Las pruebas consistieron en observar y medir, en un osciloscopio, la forma de onda de un botón aleatorio. Se debe aclarar que el modulo receptor IR utilizado tiene su salida negada, es decir, en reposo la salida es un estado “alto”. Por otro lado, no se observa la presencia de la portadora ya que el receptor posee filtros integrados al mismo. En cada caso se extrajo la siguiente información de la trama capturada de cada control remoto:

- Tiempo en “bajo” y “alto” de la cabecera.
- Tiempo en “bajo” y “alto” de un “1” lógico.
- Tiempo en “bajo” y “alto” de un “0” lógico.
- Número de bytes totales.

Esta información es la que se almacenará dentro del microcontrolador y que será consultada cuando se quiera emitir desde el dispositivo.

Reproductor DVD Tonomac

- Cabecera nivel “alto”: 4,500[ms]
- Cabecera nivel “bajo”: 9,000[ms]
- “1” lógico nivel “alto”: 1,680[ms]
- “1” lógico nivel “bajo”: 0,560[ms]
- “0” lógico nivel “alto”: 0,560[ms]
- “0” lógico nivel “bajo”: 0,560[ms]
- Cantidad de bytes: 4



Figura N° 32: Control remoto Reproductor DVD Tonomac.

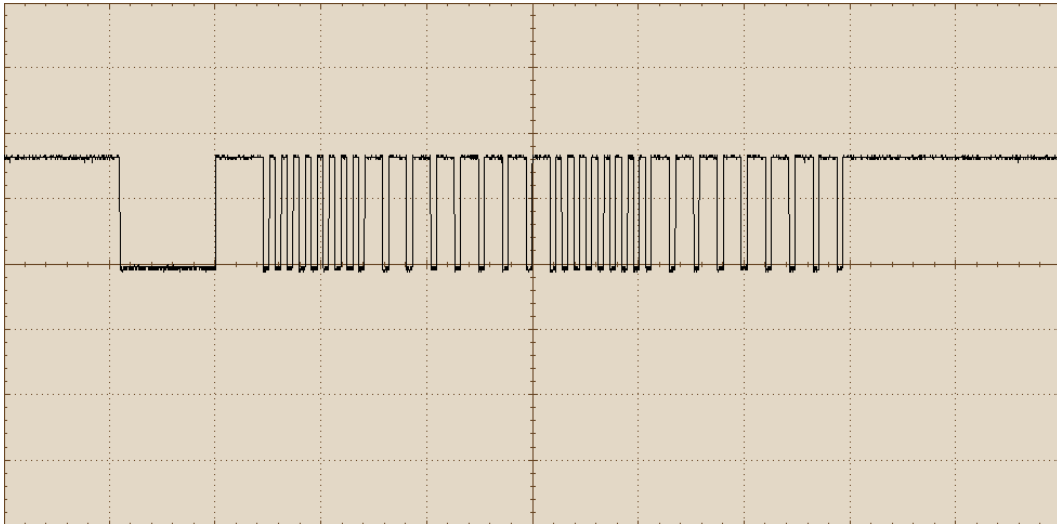


Figura N° 33: Señal control remoto Reproductor DVD Tonomac.

Decodificador RTC

- Cabecera nivel “alto”: 8,920[ms]
- Cabecera nivel “bajo”: 4,440[ms]
- “1” lógico nivel “alto”: 1,640[ms]
- “1” lógico nivel “bajo”: 0,600[ms]
- “0” lógico nivel “alto”: 0,600[ms]
- “0” lógico nivel “bajo”: 0,600[ms]
- Cantidad de bytes: 4



Figura N° 34: Control remoto Decodificador RTC.

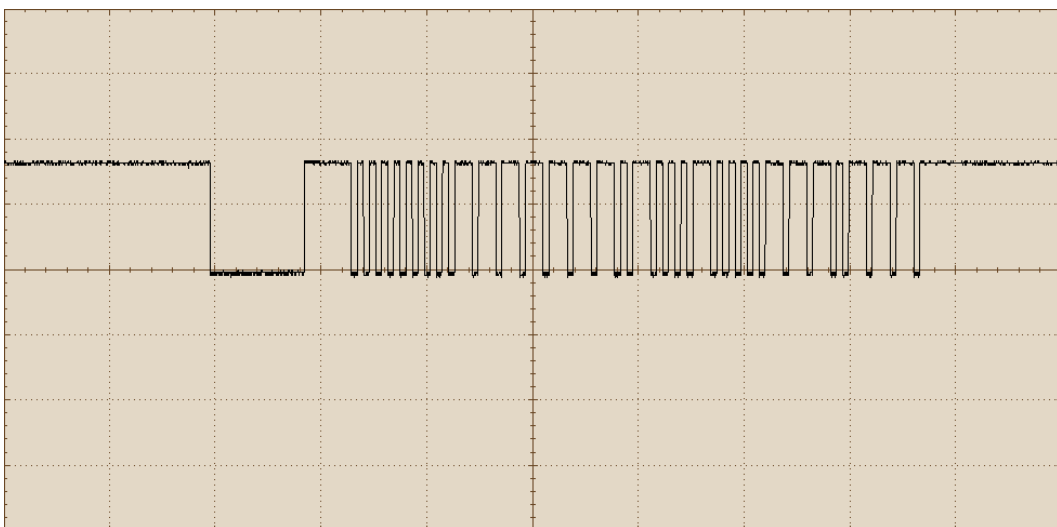


Figura N° 35: Señal control remoto Decodificador RTC.

LCD Smart TV RCA

- Cabecera nivel “alto”: 4,000[ms]
- Cabecera nivel “bajo”: 4,000[ms]
- “1” lógico nivel “alto”: 1,960[ms]
- “1” lógico nivel “bajo”: 0,480[ms]
- “0” lógico nivel “alto”: 1,000[ms]
- “0” lógico nivel “bajo”: 0,480[ms]
- Cantidad de bytes: 3



Figura N° 36: Control remoto LCD Smart TV RCA.

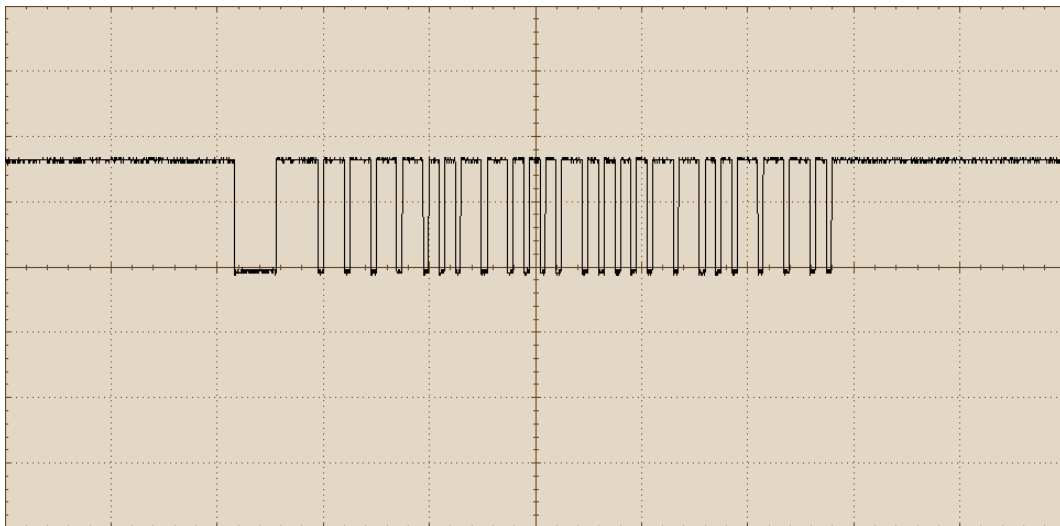


Figura N° 37: Señal control remoto LCD Smart TV RCA.

LCD BGH

- Cabecera nivel “alto”: -----
- Cabecera nivel “bajo”: -----
- “1” lógico nivel “alto”: 0,880[ms]
- “1” lógico nivel “bajo”: 0,880[ms]
- “0” lógico nivel “alto”: 0,880[ms]
- “0” lógico nivel “bajo”: 0,880[ms]
- Cantidad de bytes: -----



Figura N° 38: Control remoto LCD BGH.

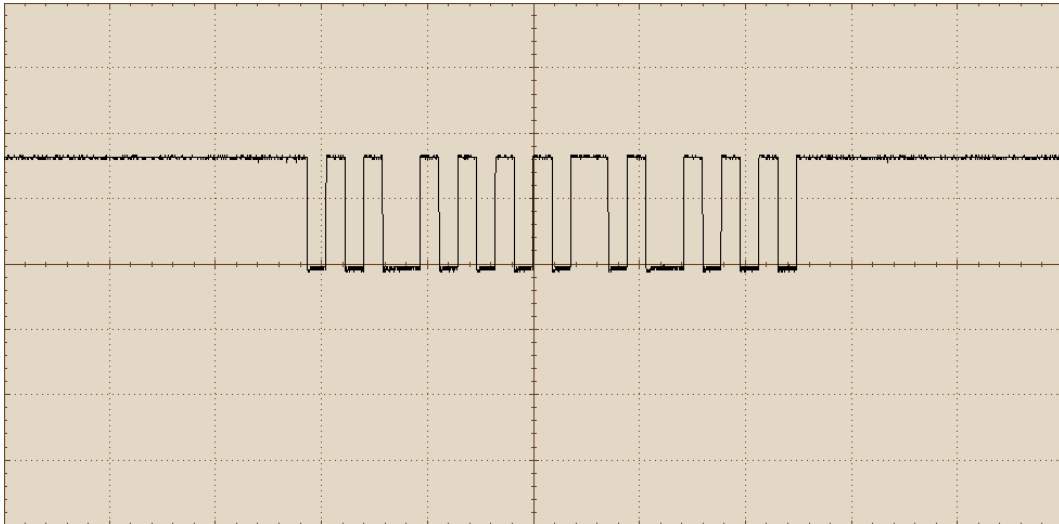


Figura N° 39: Señal control remoto LCD BGH.

LCD Noblex

- Cabecera nivel “alto”: 4,480[ms]
- Cabecera nivel “bajo”: 4,480[ms]
- “1” lógico nivel “alto”: 1,680[ms]
- “1” lógico nivel “bajo”: 0,560[ms]
- “0” lógico nivel “alto”: 0,560[ms]
- “0” lógico nivel “bajo”: 0,560[ms]
- Cantidad de bytes: 4



Figura N° 40: Control remoto LCD Noblex.

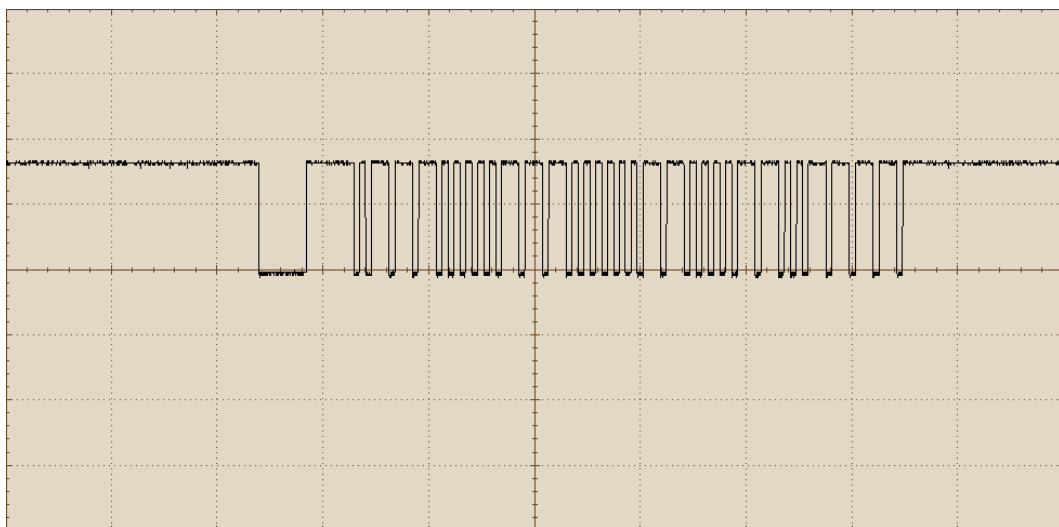


Figura N° 41: Señal control remoto LCD Noblex.

Reproductor VHS Philco

- Cabecera nivel “alto”: 8,000[ms]
- Cabecera nivel “bajo”: 4,0000[ms]
- “1” lógico nivel “alto”: 1,480[ms]
- “1” lógico nivel “bajo”: 0,480[ms]
- “0” lógico nivel “alto”: 0,480[ms]
- “0” lógico nivel “bajo”: 0,480[ms]
- Cantidad de bytes: 2



Figura N° 42: Control remoto Reprod. VHS Philco.

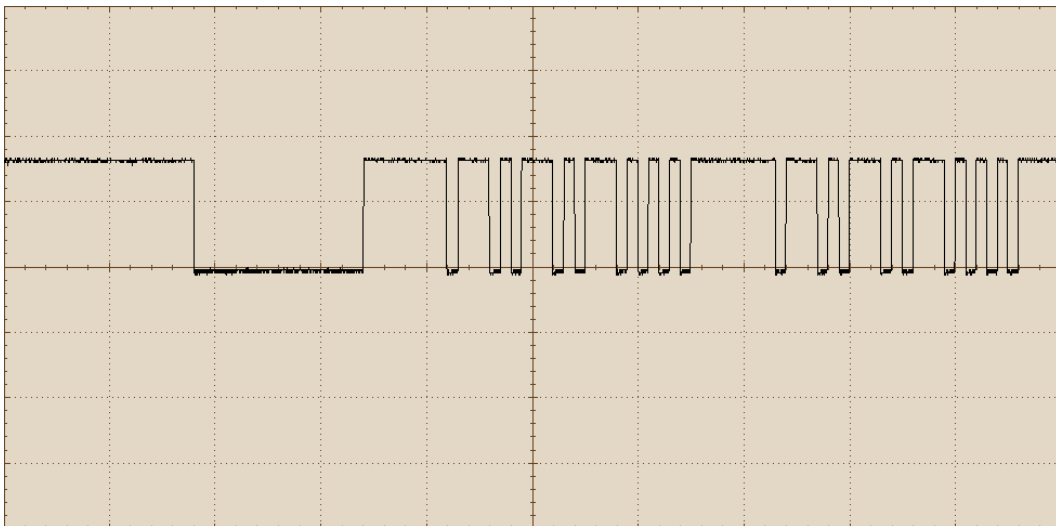


Figura N° 43: Señal control remoto Reproductor VHS Philco.

LCD Tonomac

- Cabecera nivel “alto”: 9,000[ms]
- Cabecera nivel “bajo”: 4,560[ms]
- “1” lógico nivel “alto”: 1,680[ms]
- “1” lógico nivel “bajo”: 0,600[ms]
- “0” lógico nivel “alto”: 0,600 [ms]
- “0” lógico nivel “bajo”: 0,600[ms]
- Cantidad de bytes: 4



Figura N° 44: Control remoto LCD Tonomac.

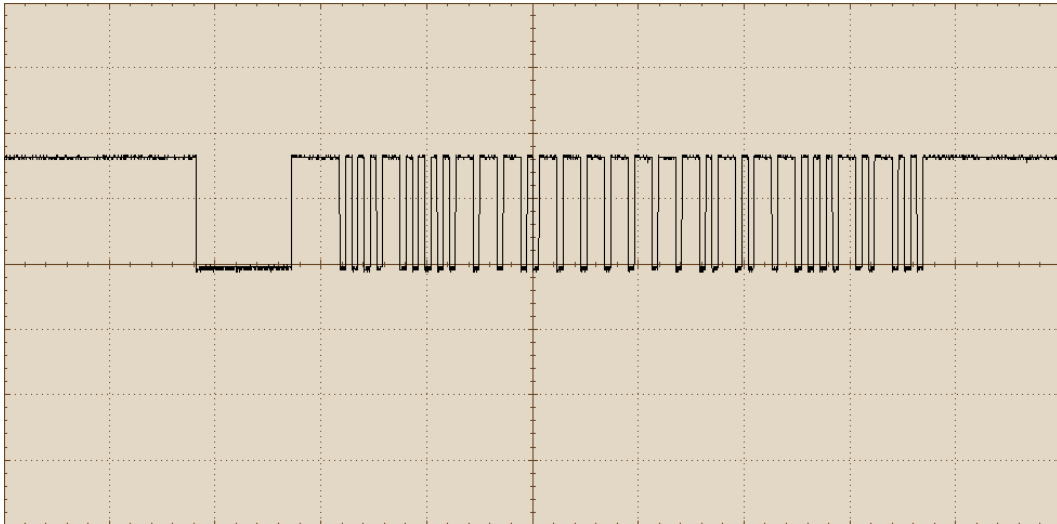


Figura N° 45: Señal control remoto LCD Tonomac.

TV Tonomac Tubo

- Cabecera nivel “alto”: 4,160[ms]
- Cabecera nivel “bajo”: 8,480[ms]
- “1” lógico nivel “alto”: 1,560[ms]
- “1” lógico nivel “bajo”: 0,560[ms]
- “0” lógico nivel “alto”: 0,560[ms]
- “0” lógico nivel “bajo”: 0,560[ms]
- Cantidad de bytes: 2



Figura N° 46: Control remoto TV Tonomac Tubo.

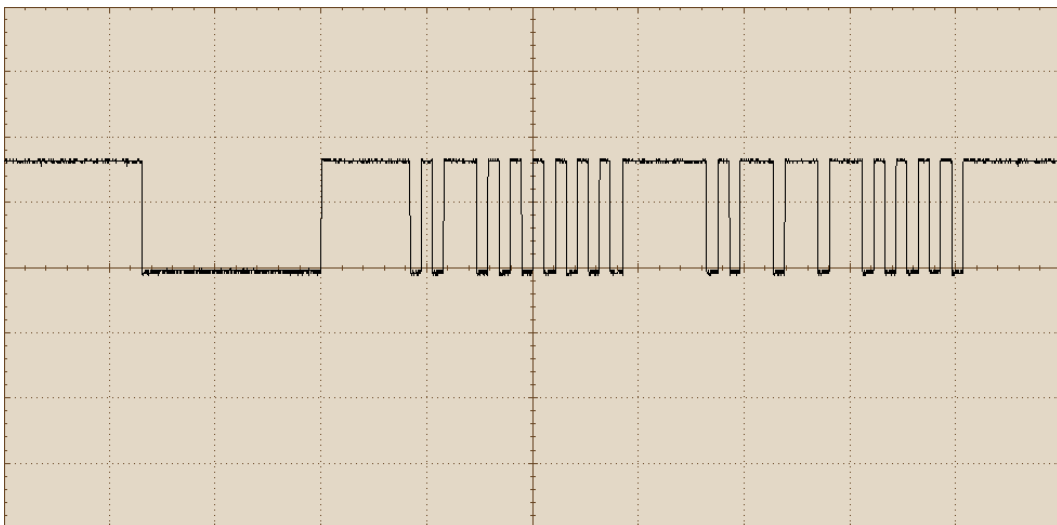


Figura N° 47: Señal control remoto TV Tonomac Tubo.

TV Philco Tubo

- Cabecera nivel “alto”: 4,000[ms]
- Cabecera nivel “bajo”: 8,000[ms]
- “1” lógico nivel “alto”: 1,480[ms]
- “1” lógico nivel “bajo”: 0,480[ms]
- “0” lógico nivel “alto”: 0,480[ms]
- “0” lógico nivel “bajo”: 0,480[ms]
- Cantidad de bytes: 2



Figura N° 48: Control remoto TV Philco Tubo.

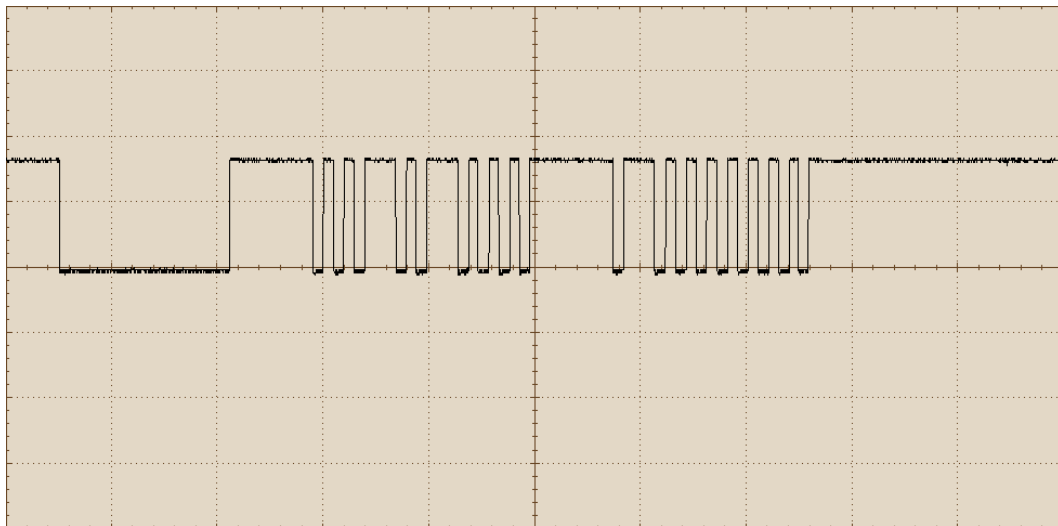


Figura N° 49: Señal control remoto TV Philco Tubo.

TV Samsung Tubo

- Cabecera nivel “alto”: 4,200[ms]
- Cabecera nivel “bajo”: 8,480[ms]
- “1” lógico nivel “alto”: 1,600[ms]
- “1” lógico nivel “bajo”: 0,560[ms]
- “0” lógico nivel “alto”: 0,560[ms]
- “0” lógico nivel “bajo”: 0,560[ms]
- Cantidad de bytes: 2



Figura N° 50: Control remoto TV Samsung Tubo.

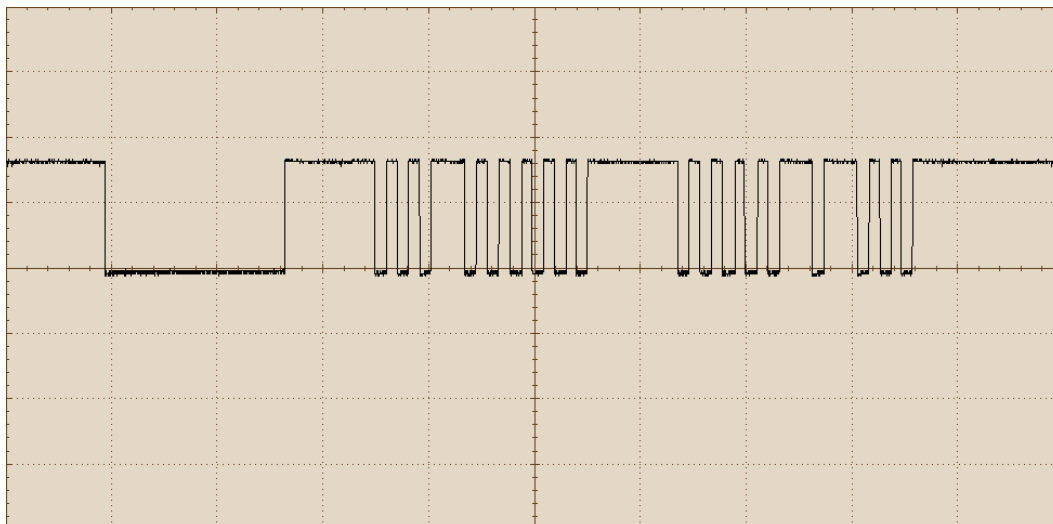


Figura N° 51: Señal control remoto TV Samsung Tubo.

A partir de información expuesta anteriormente, se puede concluir, en principio, que los distintos protocolos IR pueden ser agrupados en 3 conjuntos. Para esta clasificación se tiene en cuenta las similitudes que presentaban los protocolos.

- 1) Aquellos que poseen encabezado (AGC) y agregan de forma negada, tanto el comando como la dirección. Como es el caso del LCD Noblex.
- 2) Aquellos que poseen encabezado (AGC) pero no envían los datos negados. Como es el caso del TV Tonomac Tubo.
- 3) Aquellos que no poseen encabezado (AGC). Como es el caso del LCD BGH.

La diferencias entre protocolos de un mismo grupo son, por ejemplo, el tiempo en alto y bajo de la cabecera, de un “1” lógico, de un “0” lógico. Sin embargo, mantienen una estructura de trama muy similar. Por otro lado, el grupo 1 y 2 se diferencian que el primero envía cuatro bytes de dato mientras que en el segundo se envían dos byte de datos. Este hecho servirá, posteriormente, para unificar estos dos grupos.

2.3.6 Decodificación

Para la programación de los microcontroladores de Texas Instruments se utiliza en general el entorno de programación ofrecido por la empresa, este es el software Code Composer Studio y el lenguaje de programación es C.

El algoritmo de decodificación se basa en la medición de la cantidad de tiempo o de ciclos de reloj existentes entre los flancos, tanto ascendientes como descendientes, de la señal que genera el módulo receptor IR. Además, teniendo en cuenta los resultados de los ensayos, se han fijado umbrales de tiempos comunes a todos los protocolos ensayados. Esto quiere decir, por ejemplo, si el tiempo en alto de la señal excede a 1[ms] entonces se debe interpretar como un “1” lógico, de lo contrario se interpretará como un “0” lógico. Este proceso se realiza en tiempo real.

El proceso comienza cuando el modulo IR genera una interrupción en el microcontrolador, lo que indica que comienza una trama. En ese instante se enciende un temporizador que se incrementará hasta que ocurra el siguiente flanco. Cuando esto suceda, se detendrá el temporizador, se asignará este valor a “headerLow” y se reiniciará el temporizador desde cero. En el siguiente flanco se registrará “headerHigh” y a continuación se registrarán los “1” y “0” decidiendo entre uno y otro según si el tiempo del pulso se encuentra por encima del umbral o por debajo. Los datos decodificados se almacenaran en una estructura como la siguiente.

```
typedef struct
{
    uint32_t headerLow;
    uint32_t headerHigh;
    uint32_t tiempoUnoCero[2][2];
    uint32_t interLow;
    uint32_t interHigh;
    uint32_t stopLow;
    uint8_t datos[4];
    uint8_t numeroBytes;
    uint8_t numeroBits;
    tBoolean valido;
    uint8_t id;
} _boton;
```

Algunas de las variables que se observan en la estructura cumplen funciones auxiliares en la decodificación como, por ejemplo, “numeroBits” o “valido”. Esta información extra no será almacenada en el servidor ya que no es necesaria para poder generar la señal posteriormente.

Por otro lado, la variable “interLow” e “interHigh” almacenan los tiempos de un pulso intermedio que aparece entre el dato de “dirección” y “comando” en aquellos protocolos donde no se agrega la información negada.

El campo “id” representa una clave que identifica de forma univoca toda la información de una señal de un botón dado. Es el mismo valor con el que se identifica en el servidor.

En el caso de los equipos de aire acondicionado no se decodifican las tramas IR, es decir que el sistema no puede “aprender” a partir de una control remoto. Esto se debe a que la señal que emite el control depende del estado anterior, dicho de otro modo, un mismo botón tiene múltiples funciones. Por ejemplo, el botón MODE envía cíclicamente las órdenes de COOL, HOT, FAN, DRY, etc. Para solucionar el inconveniente anterior, se pre-cargaron en el dispositivo los códigos correspondiente a los fabricantes de aire acondicionado más comunes en el mercado.

2.3.7 Codificación

Para la generación de la señal portadora se utiliza el módulo PWM del microcontrolador, el cual se configura a una frecuencia de 38[KHz] con un ciclo de trabajo del 50%.

El método para generar la señal consiste en activar o desactivar el la señal PWM durante los intervalos temporales correspondientes. Por ejemplo, en primer lugar se activa la salida de la portadora e inmediatamente se genera un retardo correspondiente al tiempo de “headerLow”,

cuando el retardo termina se desactiva la señal PWM y se dispara un retardo equivalente a “headerHigh”, se continúa de esta manera hasta completar la trama correspondiente.

2.4 Comunicación WiFi

Un sistema WiFi es un mecanismo de conexión de dispositivos electrónicos de forma inalámbrica. Los dispositivos habilitados con WiFi (como una computadora personal, un televisor inteligente, una videoconsola, un teléfono inteligente) pueden conectarse a internet a través de un punto de acceso de red inalámbrica. Dicho punto de acceso tiene un alcance de unos veinte metros en interiores, alcance que incrementa al aire libre. WiFi es una marca de la Alianza WiFi, la organización comercial que adopta, prueba y certifica que los equipos cumplen con los estándares 802.11 relacionados a redes inalámbricas de área local.

Existen diversos tipos de WiFi, basado cada uno de ellos en una estándar IEEE 802.11 aprobado. Son los siguientes:

- Los estándares IEEE 802.11b, IEEE 802.11g e IEEE 802.11n disfrutaron de una aceptación internacional debido a que la banda de 2,4 GHz está disponible casi universalmente, con una velocidad de hasta 11 Mbit/s, 54 Mbit/s y 300 Mbit/s, respectivamente.
- En la actualidad ya se maneja también el estándar IEEE 802.11ac, conocido como WiFi 5, que opera en la banda de 5 GHz y que disfruta de una operatividad con canales relativamente limpios. La banda de 5 GHz ha sido recientemente habilitada y, además, no existen otras tecnologías (Bluetooth, microondas, ZigBee) que la estén utilizando, por lo tanto existen muy pocas interferencias. Su alcance es algo menor que el de los estándares que trabajan a 2,4 GHz (aproximadamente un 10 %), debido a que la frecuencia es mayor (a mayor frecuencia, menor alcance).

Existen otras tecnologías inalámbricas como Bluetooth que también funcionan a una frecuencia de 2,4 GHz, por lo que puede presentar interferencias con la tecnología wifi. Debido a esto, en la versión 1.2 del estándar Bluetooth por ejemplo se actualizó su especificación para que no existieran interferencias con la utilización simultánea de ambas tecnologías, además se necesita tener 40 000 kbit/s (Wikipedia, 2017).

Para lograr la conexión inalámbrica entre la aplicación móvil y la interfaz WiFi-IR se utilizó el módulo ESP8266, el cual es un componente de muy bajo costo y cuenta con las siguientes características:

- Velocidad de operación de 80[MHz].
- 64[KB] de memoria RAM para instrucciones.
- 96[KB] de memoria RAM para datos.
- 64[KB] de memoria ROM.
- Arquitectura RISC.
- Microcontrolador de 32 bits L106 de Tensilica.
- Stack protocolo TCP/IP incorporado.

- Comunicación USART con el microcontrolador principal.

En la figura N° 52 se puede observar el módulo mencionado mientras que la figura N° 53 muestra la distribución de pines, los cuales poseen las funciones descritas en la tabla N° 1.

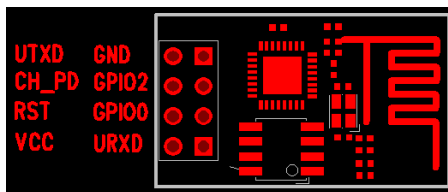


Figura N° 52: Módulo ESP8266.

Figura N° 53: Distribución de pines ESP8266.^[16]

Pin	Nombre	Descripción
1	GND	Ground
2	U0TXD	Línea transmisión UART0
3	GPIO2	Entrada/salida de propósito general
4	CH_EN	Habilitación de chip. Activo en alto
5	GPIO0	Entrada/salida de propósito general
6	RST	Reset externo. Activo en bajo.
7	U0RXD	Línea recepción UART0
8	VDD	+3,3[V]

Tabla N° 1: Descripción de pines ESP8266.

El módulo mencionado viene de fábrica con un firmware grabado tal que se puede comandar el dispositivo mediante comandos AT, lo cual es común en muchos módulos como GPRS, GSM, GPS, Bluetooth y WiFi. La figura N° 54 muestra algunos de los comandos utilizados.

Function	AT Command	Response
Working	AT	OK
Restart	AT+RST	OK [System Ready, Vendor:www.ai-thinker.com]
Firmware version	AT+GMR	AT+GMR 0018000902 OK
List Access Points	AT+CWLAP	AT+CWLAP +CWLAP:(4,"RocheFortSurLac",-38,"70:62:b8:6f:6d:58",1) +CWLAP:(4,"Lilipad2.4",-83,"f8:7b:8c:1e:7c:6d",1) OK
Join Access Point	AT+CWJAP? AT+CWJAP="SSID","Password"	Query AT+CWJAP? +CWJAP:"RocheFortSurLac" OK
Quit Access Point	AT+CWQAP=? AT+CWQAP	Query OK
Get IP Address	AT+CIFSR	AT+CIFSR 192.168.0.105 OK
Set Parameters of Access Point	AT+ CWSAP? AT+ CWSAP= <ssid>,<pwd>,<chl> , <ecn>	Query ssid, pwd chl = channel, ecn = encryption
WiFi Mode	AT+CWMODE? AT+CWMODE=1 AT+CWMODE=2 AT+CWMODE=3	Query STA AP BOTH
Set up TCP or UDP connection	AT+CIPSTART=? (CIPMUX=0) AT+CIPSTART = <type>,<addr>,<port> (CIPMUX=1) AT+CIPSTART= <id><type>,<saddr>,<port>	Query id = 0-4, type = TCP/UDP, addr = IP address, port= port
TCP/UDP Connections	AT+ CIPMUX? AT+ CIPMUX=0 AT+ CIPMUX=1	Query Single Multiple
Check join devices' IP	AT+CWLIF	
TCP/IP Connection Status	AT+CIPSTATUS	AT+CIPSTATUS? no this fun
Send TCP/IP data	(CIPMUX=0) AT+CIPSEND=<length>; (CIPMUX=1) AT+CIPSEND= <id>,<length>	
Close TCP / UDP connection	AT+CIPCLOSE=<cid> or AT+CIPCLOSE	
Set as server	AT+ CIPSERVER= <mode>[,<port>]	mode 0 to close server mode; mode 1 to open; port = port
Set the server timeout	AT+CIPSTO? AT+CIPSTO=<time>	Query <time>0~28800 in seconds
Baud Rate*	AT+CIOBAUD? Supported: 9600, 19200, 38400, 74880, 115200, 230400, 460800, 921600	Query AT+CIOBAUD? +CIOBAUD:9600 OK

Figura N° 54: Comandos AT de módulo ESP8266.^[17]

En este proyecto se optó por modificar el firmware original, de esta forma el microcontrolador propio del módulo WiFi puede realizar algunas tareas y no sobrecargar al procesador principal.

Para poder programar el ESP8266 se debe iniciarlo en modo “flash”, para esto se debe poner en un estado lógico “cero” el pin GPIO0 y luego realizar un reset del mismo. Es para ello que se incluyeron los dos pulsadores en la placa adaptadora.

A modo de comentario, además del ESP8266 también se consideró la posibilidad de utilizar un módulo RN171 de la empresa Microchip, como el que se muestra en la figura N° 55. Este componente se puede conseguir fácilmente en el país y además se cuenta con el soporte técnico por parte de las empresas proveedoras. El inconveniente que presenta este módulo es que se debe realizar el ruteo de la antena en el circuito impreso y, si bien en los manuales se describe como diseñarla, no es una tarea sencilla ya que las frecuencias de las señales involucradas son muy elevadas.



Figura N° 55: Módulo RN171.^[18]

2.4.1 Protocolo de comunicación propio

Se desarrolló un protocolo propio y simple para intercambiar información entre el módulo y la aplicación móvil, a continuación se describe el mismo.

- La aplicación envía “IR0”. Indica al sistema que a continuación se debe capturar la señal infrarroja.
 - El microcontrolador contesta “OK”.
- La aplicación envía “IR1” + id_boton. Indica al sistema que debe emitir la señal correspondiente al botón id_boton. Si el id_boton es igual a cero indica que debe emitir la señal recientemente capturada.
 - El microcontrolador contesta “OK” si tiene en su memoria los datos del botón indicado. Si no se cuenta con la información o no se hizo una captura previa, el microcontrolador contesta “ERROR”.
- La aplicación envía “IR2”: Indica al sistema que debe enviar la información recientemente capturada.
 - El microcontrolador contesta con la información decodificada en la captura de la señal. Si no se hizo una captura previa se contesta con “ERROR”.
- La aplicación envía “IR3” + información_boton. Si el microcontrolador contesta “ERROR” luego de enviar “IR1”, la aplicación envía los datos necesarios para generar la señal.

- El microcontrolador contesta “OK”.
- La aplicación envía “IR4” + información_boton_aire_acondicionado. Se envía en forma codificada la temperatura, el modo, velocidad del ventilador, fabricante y encender/apagar.
 - El microcontrolador contesta “OK”.
- La aplicación envía “IR5” + información_boton. La aplicación envía los datos necesarios para generar la señal para realizar una prueba de un botón dado.
 - El microcontrolador contesta “OK”.
- La aplicación envía “VCA” + N°_de_salida + activar/desactivar. Indica que debe activar o desactivar la salida de 220[V] indicada.
 - El microcontrolador contesta “OK”.
- La aplicación envía “SON” + activar/desactivar. Indica que se debe activar o desactivar las indicaciones sonoras.
 - El microcontrolador contesta “OK”.
- La aplicación envía “CON” + nombre_red + contraseña_red. Indica que se debe conectarse a la red indicada.
 - El microcontrolador contesta con la dirección IP que el router asignó. Si no se pudo realizar la conexión con el router se contesta “ERROR”.
- La aplicación envía “CIP”. Consulta la dirección IP actual.
 - El microcontrolador contesta con la dirección IP que el router asignó. Si no se encuentra conectado con el router se contesta “ERROR”.
- La aplicación envía “MAC”. Consulta la dirección MAC.
 - El microcontrolador contesta con la dirección MAC.

2.4.2 Protocolo de transferencia de hipertexto

HTTP es el protocolo de comunicación que permite las transferencias de información en la World Wide Web. El mismo fue desarrollado por World Wide Web Consortium e Internet Engineering Task Force, colaboración que culminó en 1999 con la publicación de una serie de RFC, el más importante de ellos es el RFC 2616 que especifica la versión 1.1. HTTP define la sintaxis y la semántica que utilizan los elementos de software de la arquitectura web (clientes, servidores, proxies) para comunicarse.

Es un protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor. El cliente realiza una petición enviando un mensaje, con cierto formato al servidor y este le envía un mensaje de respuesta. Ejemplos de cliente son los navegadores web.

El protocolo establece una serie predefinida de métodos de petición que pueden utilizarse. El protocolo tiene flexibilidad para ir añadiendo nuevos métodos y así añadir nuevas funcionalidades.

Cada método indica la acción que desea que se efectúe sobre el recurso identificado. Lo que este recurso representa depende de la aplicación del servidor. Por ejemplo el recurso puede corresponderse con un archivo que reside en el servidor. Algunos de los métodos más utilizados son: GET, POST, PUT, DELETE, HEAD (Wikipedia, 2017).

En este proyecto, el módulo ESP8266 se utiliza como un servidor mientras que la aplicación móvil funciona como un cliente realizando las peticiones HTTP correspondientes según las acciones que realiza el usuario. Todas las transferencias se realizan mediante el puerto 80.

Por ejemplo, cuando el teléfono móvil solicita la dirección MAC del módulo WiFi lo hace mediante el método GET, la respuesta por parte del ESP8266 tiene la siguiente sintaxis:

```
HTTP/1.1 200 OK
Content-Type: text/plain"
Connection: close
```

```
5C:CF:7F:16:77:08
```

Por otro lado, cuando el usuario presiona un botón de en la pantalla de su teléfono la aplicación envía una petición PUT, con el código de identificación del botón presionado. El módulo contesta de la siguiente forma:

```
HTTP/1.1 200 OK
Content-Type: text/plain"
Connection: close
```

```
OK
```

2.5 Aplicación móvil

Como interfaz de usuario se decidió utilizar una aplicación móvil compatible con el sistema operativo Android. Este sistema se encuentra presente en cientos de millones de dispositivos, y es quien domina el mercado de plataformas móviles en el mundo. Gracias a las contribuciones de la comunidad Linux y más de 300 socios de hardware, software y portátiles, Android se ha convertido rápidamente en el SO móvil de más rápido crecimiento.

La flexibilidad de Android ha hecho que sea uno de los sistemas favorito para los consumidores y desarrolladores por igual, impulsando un fuerte crecimiento en el consumo de aplicaciones. Los usuarios de Android descargan miles de millones de aplicaciones y juegos de Google Play cada mes.

Este SO está continuamente presionando los límites de hardware y software hacia adelante para traer nuevas capacidades a los usuarios y desarrolladores. Para los desarrolladores, la innovación de Android le permite crear aplicaciones potentes y diferenciadas que utilizan las últimas tecnologías móviles.

Android ofrece herramientas para crear aplicaciones que se ven bien y aprovechar las capacidades de hardware disponibles en cada dispositivo. Se adapta automáticamente a la interfaz de usuario

para obtener el máximo rendimiento en cada dispositivo, al tiempo que le da el máximo control que desee sobre su interfaz de usuario en diferentes tipos de dispositivos.

Por ejemplo, puede crear un único binario de aplicación optimizado para los factores de forma del teléfono y de la tableta. Usted declara su interfaz de usuario en conjuntos ligeros de recursos XML, un conjunto de partes de la interfaz de usuario que son comunes a todos los factores de forma y otros conjuntos de optimizaciones específicas para teléfonos o tabletas. En tiempo de ejecución, Android aplica los conjuntos de recursos correctos en función de su tamaño de pantalla, densidad, entorno regional y así sucesivamente.

Para el desarrollo de aplicaciones en Android se ofrece un IDE completo de Java con funciones avanzadas para desarrollar, depurar y empaquetar aplicaciones de Android. Mediante el IDE, puede desarrollarse sobre cualquier dispositivo Android disponible o crear dispositivos virtuales que emulen cualquier configuración de hardware.

Por otro lado, Google Play es el principal mercado para vender y distribuir aplicaciones de Android. Como un mercado abierto, Google Play otorga el control de cómo vender los productos. Se puede publicar cuando se desee, con la frecuencia y con los clientes que desee. Puede distribuirse ampliamente a todos los mercados y dispositivos o centrarse en segmentos específicos, dispositivos o rangos de capacidades de hardware (Sitio web oficial Android).

Luego de la introducción al sistema operativo Android realizada anteriormente, se procede a describir las funciones y el uso de la aplicación diseñada. La misma se realizó mediante el entorno de desarrollo oficial conocido como Android Studio. La apariencia inicial de la aplicación, luego de instalarla, es la que se muestra en la figura N° 56.

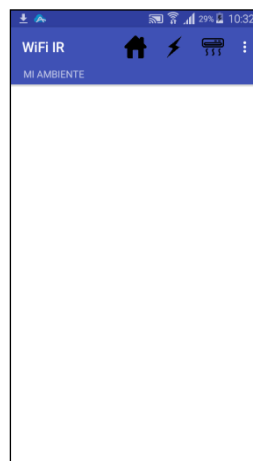


Figura N° 56: Pantalla inicial de la aplicación.

En el menú principal, ubicado en la “toolbar” se encuentran los principales controles, estos son:

- **Seleccionar ambiente:** Este botón despliega una lista con los ambientes configurados por el usuario y permite seleccionar uno de ellos. El nombre del ambiente actual aparece en la parte superior de la pantalla principal.
- **Control de cargas AC:** Este botón muestra la pantalla que permite activar o desactivar las salidas de corriente alterna.
- **Control de aire acondicionado:** Este botón muestra la pantalla donde se puede comandar los

equipos de aire acondicionado.

- Cargar botones: Este botón despliega el formulario y comandos para subir al servidor la información de un nuevo botón.
- Ver botones: Esta acción permite ver los botones que se encuentran actualmente disponibles en el servidor.
- Editar pantalla: Este botón habilita el modo “editor”, en el cual se puede agregar y ubicar los botones virtuales en cada ambiente. A este modo se puede acceder, además, haciendo doble clic en la pantalla.
- Ajustes: Este botón muestra la pantalla donde se realizan las configuraciones generales del sistema.

En cuanto al control de aire acondicionado, se implementaron las acciones para comandar equipos de los fabricantes TCL, ICP RESIDENTIAL e HITACHI. En futuras actualizaciones se agregarían otros fabricantes. Cabe mencionar que todas las actualizaciones se realizarían a través de la tienda de Google, de forma remota y automatizada. En las figuras N° 57, N° 58, N° 59 y N° 60 se muestran algunas de las pantallas que conforman la aplicación.

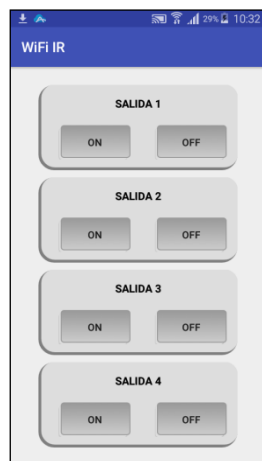


Figura N° 57: Pantalla de control de cargas AC.

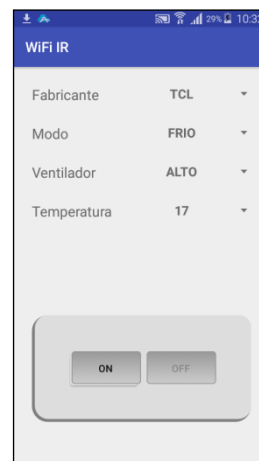


Figura N° 58: Pantalla de control aire acondicionado.



Figura N° 59: Pantalla de “modo editor”.

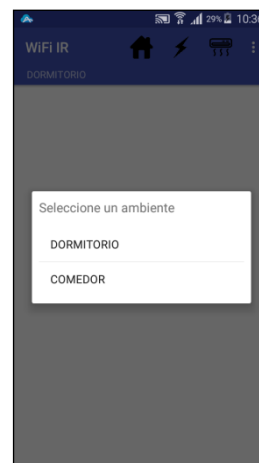


Figura N° 60: Pantalla de selección de ambiente

La figura N° 61 muestra la pantalla para subir un nuevo botón, de un control remoto dado, al servidor. En primer lugar se debe presionar el botón “CAPTURAR”, en ese momento la aplicación envía la orden al sistema para que capture la siguiente señal infrarroja, el usuario tiene 5 segundos para presionar el botón del control remoto que quiere almacenar. Luego, debe presionar “PROBAR” para comprobar que el sistema decodificó de forma correcta. Si la prueba es exitosa, el usuario debe completar los datos del formulario, donde la descripción y la imagen son opcionales. Por último se deberá presionar el botón “GUARDAR” para que la aplicación almacene toda la información en el servidor.

En la figura N° 62 se observa la pantalla de configuraciones. Desde allí, se puede activar o desactivar los avisos sonoros y de vibración, realizar la configuración manual de los ambientes e indicar al dispositivo la red WiFi a la cual debe conectarse.

El método para configurar y asignar uno de los dispositivos a un ambiente en particular es como se describe a continuación:

- 1) Se debe presionar un switch ubicado convenientemente en la placa del dispositivo, esto provocará que el mismo se configure en modo “Access point” creando una red propia llamada “WiFi IR”.
- 2) Presionar sobre “Nombre de la red”. La aplicación detectará las redes disponibles en el lugar y las mostrará en pantalla. Se deberá seleccionar la red a la cual se conectará el dispositivo.
- 3) Presionar sobre “Contraseña de la red” e ingresar la contraseña de la red anteriormente seleccionada.
- 4) Presionar “Conectar”. La aplicación indicará al dispositivo que se conecte a la red indicada, si la conexión es exitosa se mostrará una lista con los ambientes previamente añadidos en la aplicación. El usuario seleccionará a que ambiente pertenece el dispositivo, pudiendo añadir en ese momento uno nuevo si es necesario.
- 5) El dispositivo utiliza una asignación dinámica de IP, por lo que se recomienda configurar el router para que se reserve dicha IP para el dispositivo en cuestión, esto asegura que cuando se desconecte o se quite la alimentación se asigne la misma dirección al reconectarse.



Figura N° 61: Pantalla de carga de un botón.

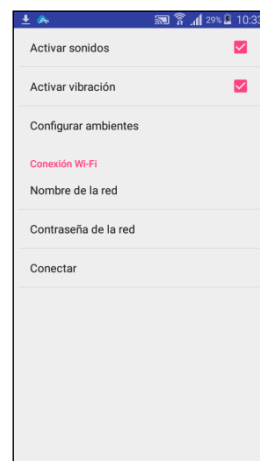


Figura N° 62: Pantalla de ajustes.

Además del menú principal, existe un menú contextual que puede ser accedido manteniendo pulsado algunos de los “botones virtual”. La figura N° 63 muestra el menú mencionado, donde las opciones son:

- Asignar nombre: permite asignar un nombre al botón en cuestión.
- Asignar función: permite asignar una función de las disponibles en el servidor.
- Ver función: permite ver que función tiene asignada el botón actualmente.
- Eliminar: permite eliminar el botón en cuestión.

En la figura N° 64 se observa un control remoto virtual completo. En la parte superior se encuentran los botones que corresponden a un televisor, mientras que en la parte central los botones comandan un equipo de música, en particular las funciones de la radio. Por último, en la parte inferior se ubican los botones que controlan el equipo reproductor de DVD.

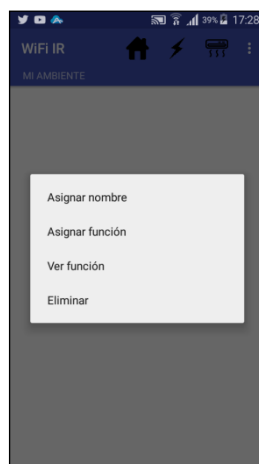


Figura N° 63: Pantalla menú contextual.

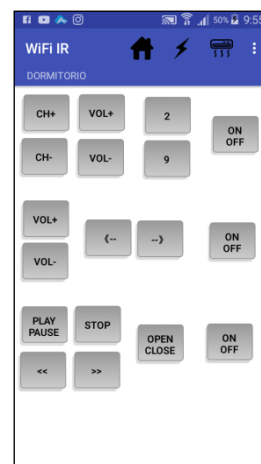


Figura N° 64: Pantalla del control completo.

2.6 Servidores

Con el fin de que la información capturada se las señales infrarrojas esté disponible y accesible para ser utilizada por cualquier usuario que lo necesite, se hizo uso de un servidor, en este caso gratuito, que puede ser accedido a través de internet.

Un servidor, como la misma palabra indica, es un ordenador o máquina informática que está al “servicio” de otras máquinas, ordenadores o personas llamadas clientes y que le suministran a estos, todo tipo de información. Por tanto, se tiene el esquema representado en la figura N° 65, conocido comúnmente como “cliente-servidor”, es uno de los más usados ya que en él se basa gran parte de internet.

Como vemos, tenemos una máquina servidora que se comunica con variados clientes, todos demandando algún tipo de información. Esta información puede ser desde archivos de texto, video, audio, imágenes, emails, aplicaciones, programas, consultas a base de datos, etc.

Por regla general, las máquinas servidoras suelen ser algo más potentes que un ordenador normal.

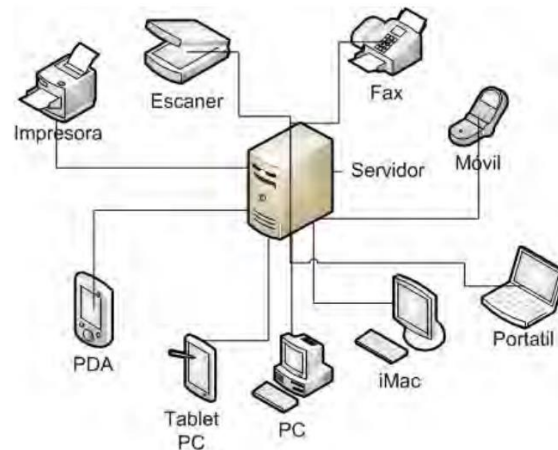


Figura N° 65: Esquema de un servidor.^[9]

Sobre todo suelen tener más capacidad tanto de almacenamiento de información como de memoria principal, ya que tienen que dar servicio a muchos clientes. Pero como todo, también depende de las necesidades, ya que podemos tener un servidor de menores prestaciones si vamos a tener pocos clientes conectados, o si los servicios que queramos en el servidor no requieren una gran capacidad servidora. A modo de ejemplo, podríamos hacer funcionar un ordenador en nuestra casa como si fuera un servidor, aunque esto no es lo más habitual. Por general, los servidores suelen estar situados en centros de datos de empresas (edificios con grandes salas dedicadas a alojar a los servidores)(Sierra García, 2006).

2.6.1 Servidor JSON

Para el desarrollo de este prototipo se utilizaron, por conveniencia y simplicidad de uso, los servicios prestados por el sitio web *myjson.com*. Este servidor permite crear archivos en formato JSON a través de un navegador web y luego leer o escribir los mismos por medio de dispositivos, como teléfonos móviles, computadoras o un módulo WiFi como el que se utiliza aquí.

JSON es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y generarlo. Está basado en un subconjunto del Lenguaje de Programación JavaScript, Standard ECMA-262 3rd Edition - Diciembre 1999. JSON es un formato de texto que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, y muchos otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

JSON está constituido por dos estructuras:

- Una colección de pares de nombre/valor. En varios lenguajes esto es conocido como un objeto, registro, estructura, diccionario, tabla hash, lista de claves o un arreglo asociativo.
- Una lista ordenada de valores. En la mayoría de los lenguajes, esto se implementa como arreglos, vectores, listas o secuencias.

Estas son estructuras universales; virtualmente todos los lenguajes de programación las soportan de una forma u otra. Es razonable que un formato de intercambio de datos que es independiente del lenguaje de programación se base en estas estructuras (Sitio web json.org).

```
{
  "nombre" : "Adrian",
  "apellido" : "Paredes",
  "cantHijos" : 0,
  "soltero" : false,
  "fechaNacimiento" : {
    "year" : 82,
    "month" : 5,
    "day" : 4,
    "date" : 3,
    "hours" : 0,
    "minutes" : 0,
    "seconds" : 0,
    "time" : 391921200000,
    "timezoneOffset" : 180,
  }
}
```

Figura N° 66: Formato JSON.^[19]

En este proyecto se utilizaron dos archivos JSON. El primero de ellos almacena los datos necesarios para generar una señal infrarroja de un botón y control remoto determinado. Por otro lado, el segundo archivo almacena el nombre, fabricante, modelo y tipo de dispositivo del mismo botón anterior. Para relacionar la información entre los dos archivos se utiliza un número único de identificación. A continuación se muestra los primeros dos objetos del primer archivo mencionado anteriormente, debe notarse que el nombre de cada objeto es precisamente el ID.

```
"1": {
  "headerHigh": "76008",
  "headerLow": "150158",
  "tiempoUnoHigh": "28013",
  "tiempoUnoLow": "8994",
  "tiempoInter": "0",
  "dato0": "16",
  "dato1": "239",
  "dato2": "1",
  "dato3": "254"
},
"2": {
  "headerHigh": "75610",
  "headerLow": "150101",
  "tiempoUnoHigh": "28031",
  "tiempoUnoLow": "9398",
  "tiempoInter": "0",
  "dato0": "16",
  "dato1": "239",
  "dato2": "113",
  "dato3": "142"
}
```

A continuación se muestra, parcialmente, el segundo archivo en donde se observa la información de los botones correspondientes al ID 1 e ID 2.

```
{
  "nombre": "Dispositivos",
  "items": [
    {
```

```
"nombre": "Televisor",
"items": [
  {
    "nombre": "TONOMAC",
    "items": [
      {
        "items": [
          {
            "nombre": "Subir canal",
            "idir": "1"
          },
          {
            "nombre": "Bajar canal",
            "idir": "2"
          }
        ]
      },
      {
        "nombre": "TO-39-4422-LED",
        "descripcion": "LED 39\""}
    ]
  }
]
```

En este proyecto se definió que la comunicación con el servidor se realice a través de la aplicación móvil, de esta forma se canaliza las transmisiones y se puede detectar con mayor facilidad fallos de conexión o similares. Esto surgió a partir de las primeras pruebas en donde el módulo WiFi se comunicaba directamente con el servidor para gestionar uno de los archivos JSON, mientras que el restante lo procesaba el teléfono móvil. Sin embargo, si alguno de los dos fallaba al subir o bajar la información, en determinados casos, no lograban notificarse el error entre sí y se perdía la sincronización de los mismos. Por otro lado, el módulo WiFi posee memoria limitada en comparación con un teléfono móvil, esto obligaba a dividir el archivo del servidor en múltiples archivos de menor tamaño y se debía determinar cuál de ellos descargar a partir del ID solicitado para emitir. Todo esto hacía que el procesamiento sea de mayor complejidad, contribuía a aumentar la posibilidad de errores y es por ello que se decidió que sea la aplicación móvil quien inicia y gestiona todos los intercambios de información.

2.6.2 Servidor de imágenes

Se hizo uso de un servidor de imágenes para almacenar y recuperar las fotografías de los dispositivos de los cuales se encuentran disponibles uno o más botones en el servidor JSON. En este caso se utiliza la plataforma *cloudinary.com* la cual, además de ofrecer almacenamiento de imágenes, provee servicios para procesar las mismas en forma online, esto es, se sube una imagen dada y, previo a descargarla, se puede solicitar que se le aplique algún recorte, o un filtro, o modificar el brillo y contraste o aplicar procesos más complejos como detectar un rostro y centrar la imagen en el mismo.

La imagen se sube al servidor en el momento en que el usuario presiona “GUARDAR” en el formulario de la figura N° 61, anteriormente mostrada. Previamente se debe presionar sobre la imagen por defecto que se presenta en la parte inferior, para lanzar la galería del teléfono móvil y poder seleccionar la fotografía deseada.

Por otro lado, cuando el usuario se encuentra visualizando la lista de modelos de un dispositivo y fabricante dado, puede mantener presionado sobre un ítem de la lista y se mostrará una pantalla con la descripción y la imagen del dispositivo en cuestión, similar a la que se observa en la figura N° 67.

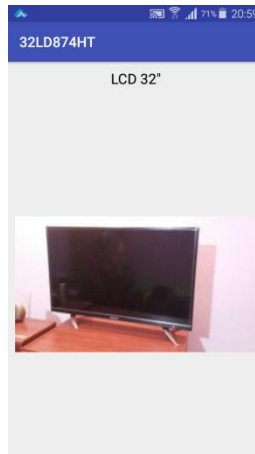


Figura N° 67: Pantalla descripción de modelo de un dispositivo.

2.7 Control de cargas AC

Se diseñó y ensayó un circuito para realizar control encendido/apagado sobre cargas de corriente alterna de 220[V]. En la figura N° 68 se visualiza el circuito utilizado. Para realizar las pruebas y por comodidad se implementó el mismo en placa experimental, tal como se observa en las figuras N° X y N° X, en donde se utilizó un switch para simular la señal proveniente del microcontrolador.

El TRIAC escogido es el L0109N, el cual soporta una corriente RMS de hasta 1[A], una corriente pico de hasta 12[A] y una tensión de bloque de hasta 800[V]. Estas características son más que suficiente para controlar iluminación y ventiladores pequeños. Los ensayos se realizaron sobre lámparas incandescentes, halógenas y de bajo consumo, en todos los caso el sistema se comportó de forma correcta. Como *driver* del TRIAC se utilizó el optoacoplador MOC3021M, el cual posee una tensión de trabajo de hasta 400[V] y permite aislar los circuitos lógicos de comando de la carga de corriente alterna. Además, se incluyó un circuito supresor o *snubber* el cual tiene la función de reducir el estrés eléctrico del TRIAC cuando se utilizan cargas fuertemente inductivas.

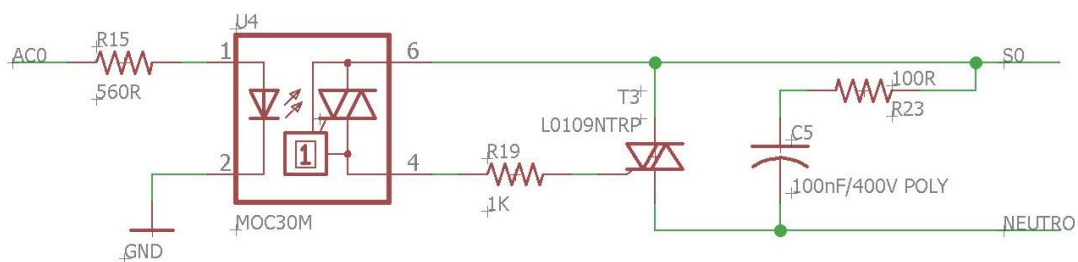


Figura N° 68: Circuito control AC.

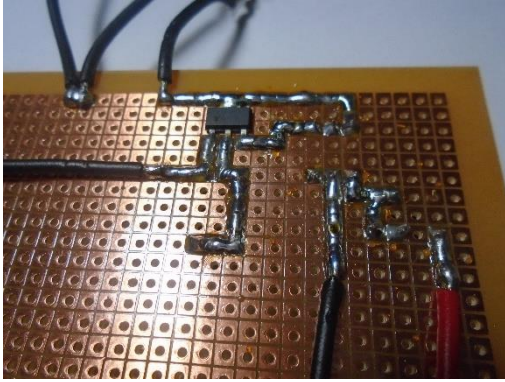


Figura N° 69: Implementación control AC (Bottom).

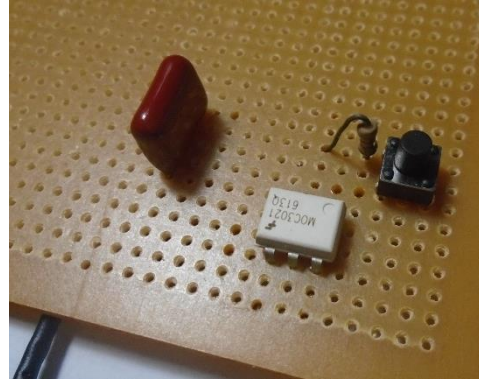


Figura N° 70: Implementación control AC (Top).

2.8 Fuente de alimentación

Como fuente de poder se intentó utilizar una batería de teléfono celular de Li-Ion. Los ensayos se realizaron sobre una batería BC50 de Motorola, con poco tiempo de uso. La figura N° 71 muestra la misma.

Figura N° 72: Batería BC50 Motorola.^[20]

Además, se implementó un cargador de baterías basado en el circuito integrado BQ24090 de la compañía Texas Instrument, el cual posee las siguientes características:

- Corriente de pre-carga y de terminación configurable.
- Selección de límite de corriente de entrada, entre 100[mA] y 500[mA].
- Protección de sobretensión sobre la batería.
- Protección de exceso de temperatura, monitoreando el NTC propio de la batería.
- Temporizador interno de seguridad.
- Indicadores led de estado de carga.

El circuito desarrollado es el que se muestra en la figura N° 73, el cual se corresponde con el sugerido en la hoja de datos del componente. Con los valores de resistencia seleccionados se fija 20[mA] como corriente de carga inicial, 10[mA] de terminación y una corriente de carga rápida de 200[mA].

En las figura N° 74 se observa la representación 3D de la placa diseñada, mientras que la figura N° 75 muestra la misma fabricada.

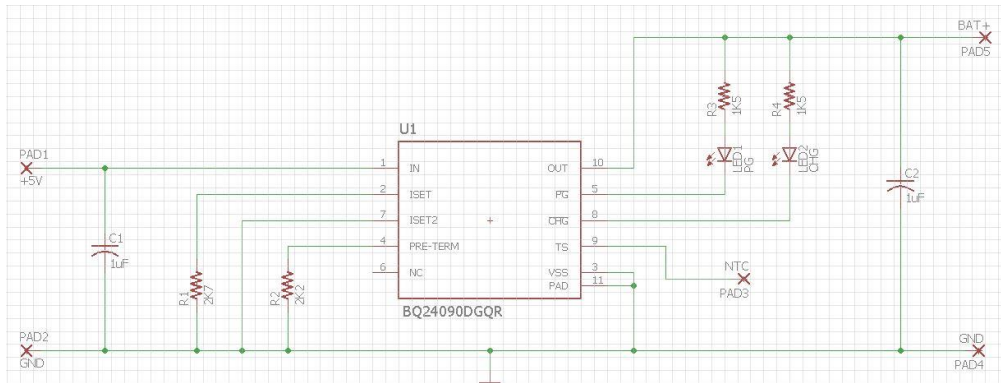


Figura N° 73: Circuito cargador de baterías.

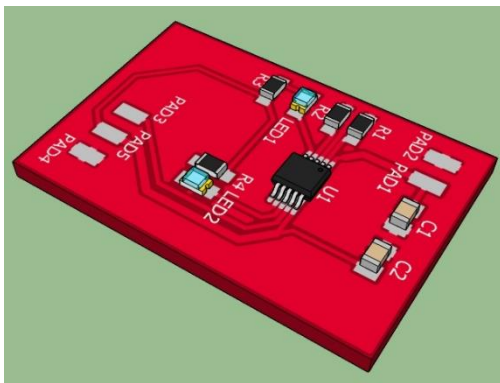


Figura N° 74: Placa 3D cargador.

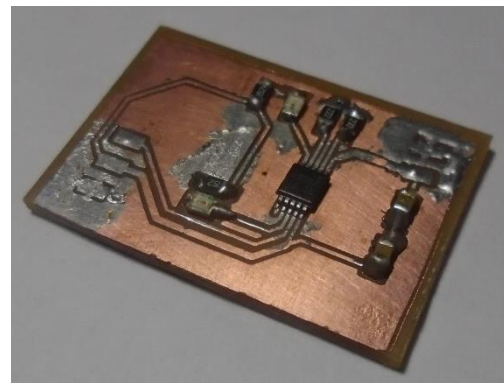


Figura N° 75: Placa real cargador.

Con las pruebas realizadas se logró observar que el cargador funciona de manera adecuada, coincidiendo las mediciones de tensión y corrientes con las calculadas en la etapa de diseño. Sin embargo, cuando se conecta el sistema a la batería, el valor de tensión de la misma se reduce llegando muy cerca del valor límite de entrada del regulador de tensión de 3,3[V]. Esto es un inconveniente ya que la autonomía resultaría de muy poco tiempo con lo cual no tendría utilidad el uso de una batería.

Para el desarrollo de este prototipo se optó por suprimir la batería y el cargador, y alimentar el sistema directamente con un cargador de teléfono móvil. Sin embargo, la placa deberá contar con un regulador de 3,3[V], en este caso se utilizará el circuito integrado TPS76733 de Texas Instruments, el cual soporta una corriente de hasta 1[A] con una caída de tensión de 230[mV].

2.9 Diseño completo

2.9.1 Hardware

Tal como se puede observar en las páginas anteriores, todo el desarrollo del proyecto se llevó a cabo implementando cada etapa de forma separada, fabricando placas parciales y realizando los ensayos pertinentes. Esto facilitó la detección previa de fallas e inconvenientes, y por consiguiente permitió corregirlas oportunamente. De este modo se logró arribar con seguridad, a la etapa de diseño de la placa definitiva en la cual convergen todos los bloques descriptos.

Igual que en los otros casos, la placa se diseñó mediante el software EAGLE Professional Edition y la fabricación de la misma fue realizado por la compañía CADIPEL S.A., ubicada en la Ciudad Autónoma de Buenos Aires. Esta empresa tiene la particularidad de estar focalizada en el desarrollo rápido de prototipos funcionales. Las características de fabricación que se ofrece son las siguientes:

Características generales:

- Materiales: FR4 y MCPCB (Metal Core PCB)
- Espesor: 1,6 mm.
- Dimensiones máximas: 280mm. x 340mm.
- Layers: 1 o 2 con PTH (Plated Through Hole)
- Espesor del cobre: 35um.
- Terminación: Cobre + OSP (Organic Solderability Preservative)
- Máscara antisoldante: Fotoimageable color verde
- Serigrafía de componentes: No

Características del diseño:

- Ancho de pista mínimo: 0,2mm. (8 mils)
- Separación mínima entre pistas: 0,2mm. (8 mils)
- Diámetro mínimo de agujero: 0,4mm. (16 mils)
- Diámetro mínimo de pad o vía: 0,8mm. (32 mils)
- Distancia de cobre a borde: 0,3mm. (12 mils)

Desde la figura N° 75 a N° 82 se muestra la placa diseñada, fabricada y montada.

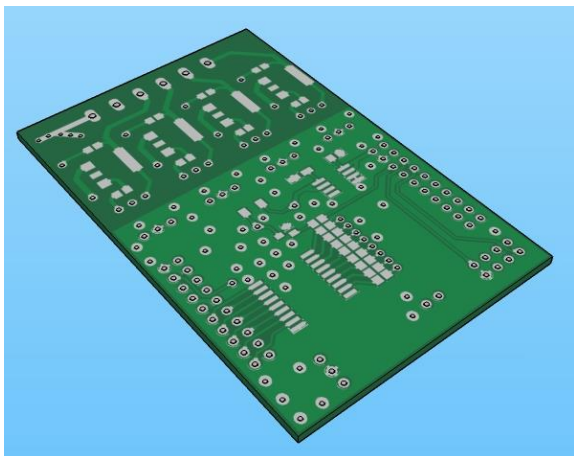


Figura N° 75: Placa definitiva 3D (bottom).

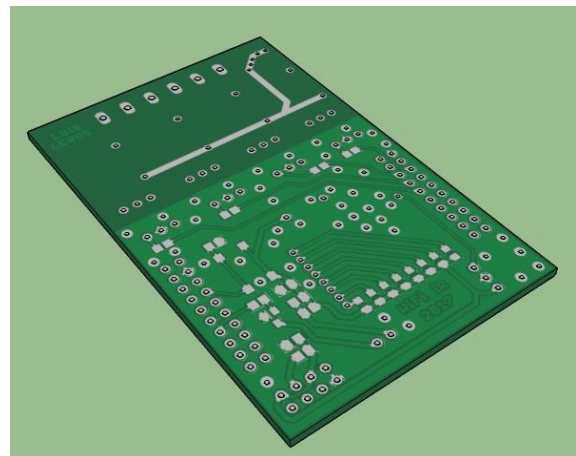


Figura N° 76: Placa definitiva 3D (top).

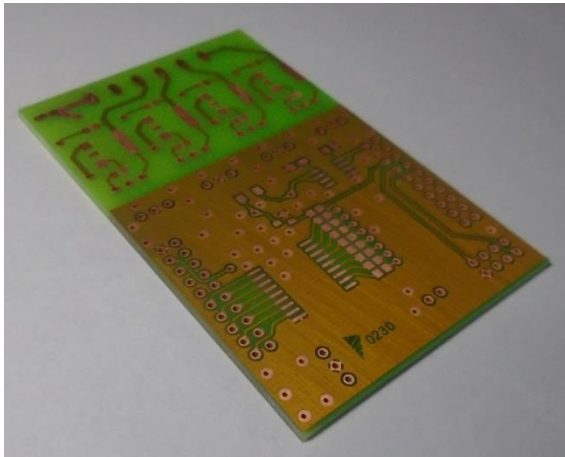


Figura N° 77: Placa definitiva real (bottom).

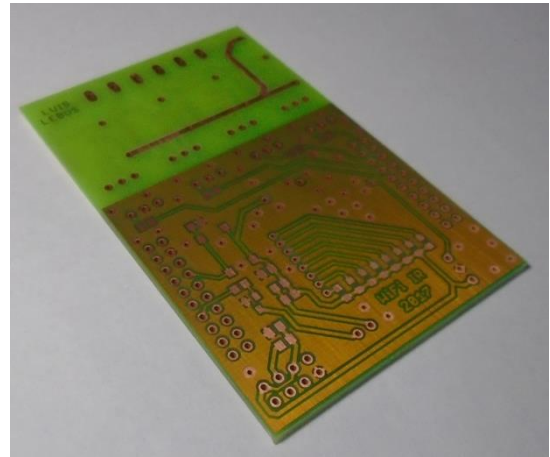


Figura N° 78: Placa definitiva real (top).

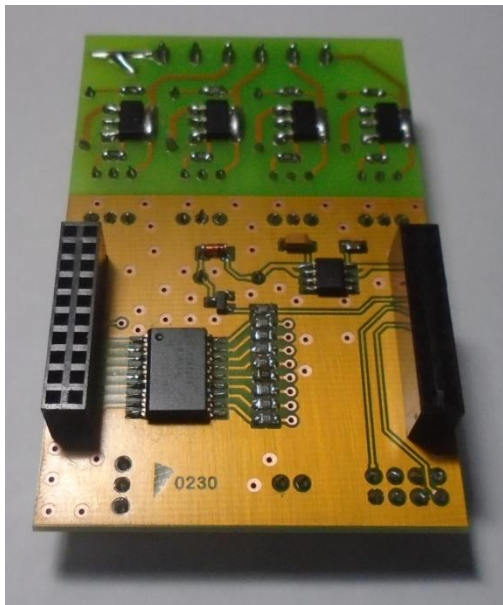


Figura N° 79: Placa definitiva real montada (bottom).

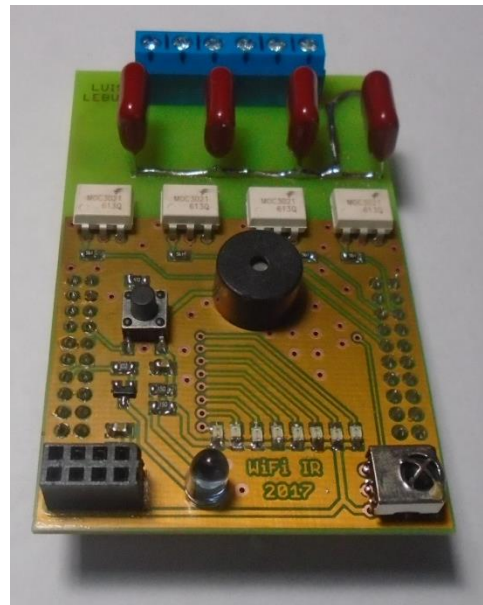


Figura N° 80: Placa definitiva real montada (top).

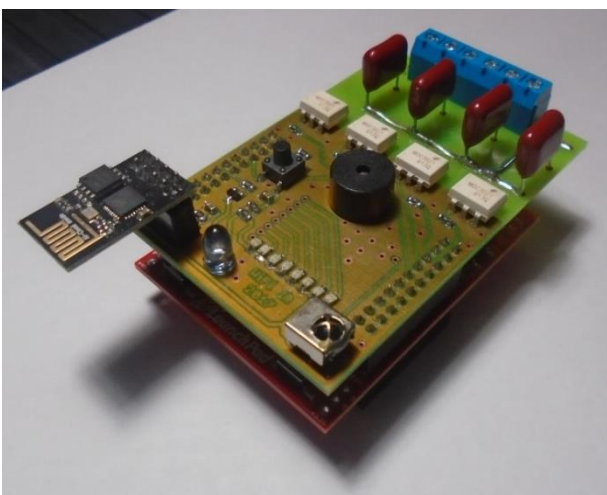


Figura N° 81: Placa definitiva acoplada a Stellaris (der).

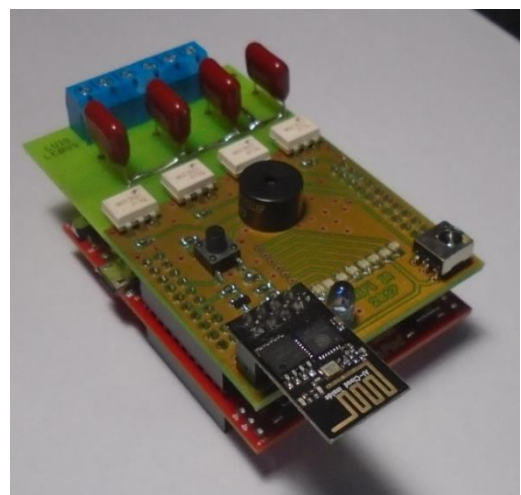


Figura N° 82: Placa definitiva acoplada a Stellaris (izq).

En las figuras N° 83 y N° 84 se muestran los esquemáticos definitivos para este prototipo, en donde se pueden hacer las siguientes observaciones:

- Se incorporó un conjunto de 8 *leds*, de montaje superficial y color rojo, junto al circuito integrado ULN2803A el cual funciona como *buffer* de corriente. El significado de cada *led* es como se describe a continuación, de izquierda a derecha según la ubicación en la placa:
 1. Se enciende cuando el sistema esta energizado e inicializado.
 2. Se enciende si el sistema se encuentra conectado al *router*, en caso contrario permanece intermitente.
 3. Se enciende por un instante cada vez que llega una petición por parte de la aplicación móvil. Además, cuando se necesita capturar una señal infrarroja este *led* parpadea indicando que se debe presionar el botón a registrar.
 4. Realiza un parpadeo cuando ocurre un error como, por ejemplo, cuando falla la decodificación.
 5. Se enciende o apaga según si la primera salida AC esta activada o desactivada.
 6. Se enciende o apaga según si la segunda salida AC esta activada o desactivada.
 7. Se enciende o apaga según si la tercera salida AC esta activada o desactivada.
 8. Se enciende o apaga según si la cuarta salida AC esta activada o desactivada.
- Se incorporó un pulsador el cual tiene dos funciones:
 1. Cuando se mantiene presionado por más de 1 segundo se elimina de la memoria, tanto RAM como EEPROM, la información almacenada sobre las señales de los últimos botones emitidos. Esta función es una forma de realizar un *reset* del sistema y puede ser de utilidad en determinados casos, como cuando se ha decodificado de forma incorrecta o se presenta algún comportamiento incorrecto.
 2. Si se presiona por menos de 1 segundo, el módulo WiFi se configura como *Access Point*. Esto es necesario cuando se quiere establecer el ambiente del hogar en el que se sitúa el dispositivo.
- Se incorporó un *buzzer* que emite un aviso sonoro en distintas situaciones, como cuando la aplicación móvil realiza una petición o cuando se borra la memoria mediante el pulsador, por ejemplo. Este aviso puede activarse o desactivarse mediante el teléfono celular.

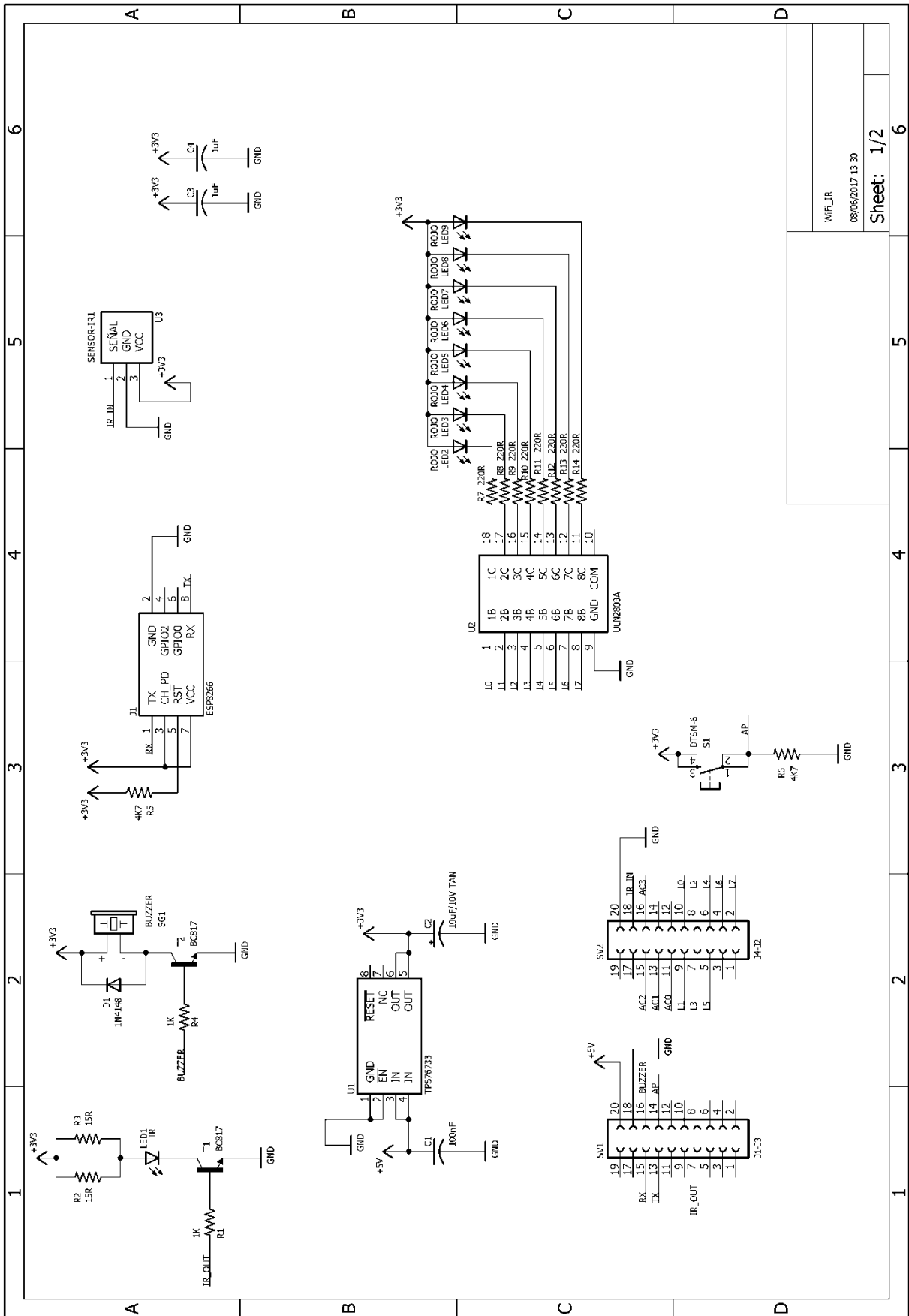


Figura N° 83: Esquemático placa definitiva (Página 1).

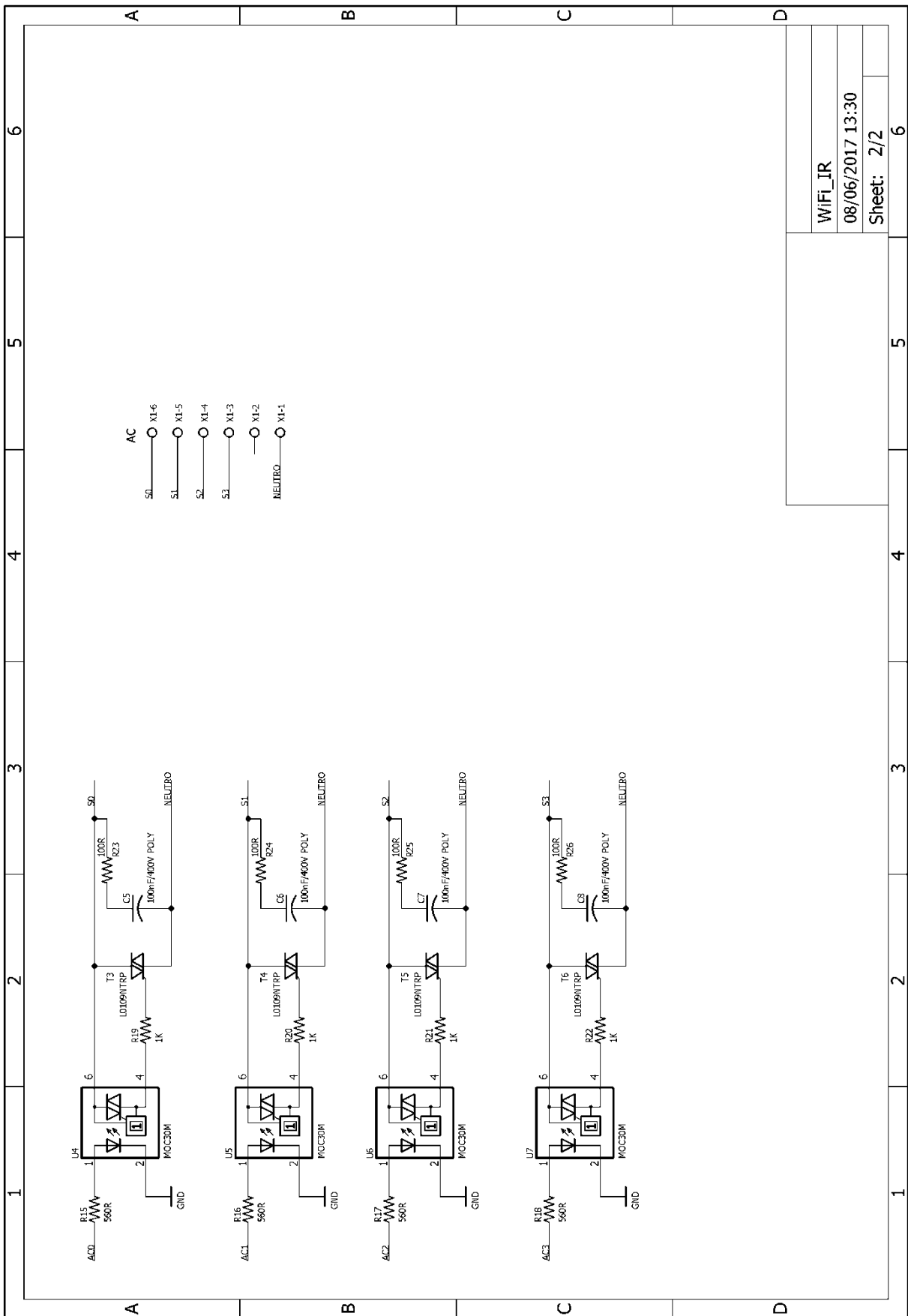


Figura N° 84: Esquemático placa definitiva (Página 2).

2.9.2 Software

Al igual que se mencionó anteriormente, el firmware se realizó mediante Code Composer Studio, el entorno oficial de la empresa Texas Instruments. En las figuras N° 85 y N° 86 se puede observar el diagrama de flujo del firmware desarrollado. El mismo se presenta, por conveniencia, de forma simplificada, ya que algunos algoritmos presentan una complejidad relativamente alta y se basan en patrones encontrados luego de múltiples ensayos realizados. Para la comprensión del mismo se debe tener presente el protocolo de comunicación propio implementado entre el sistema principal y la aplicación móvil.

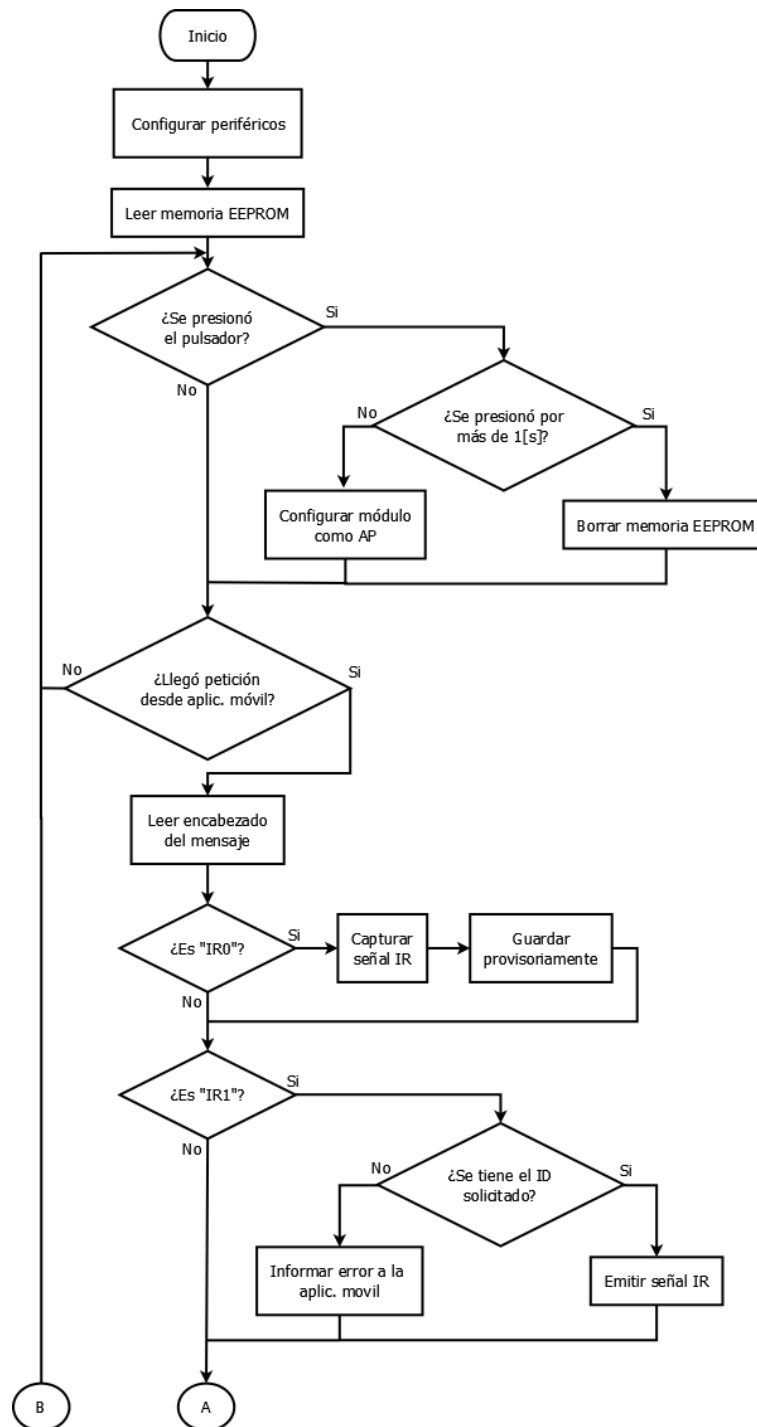


Figura N° 85: Diagrama de flujo de firmware (1).

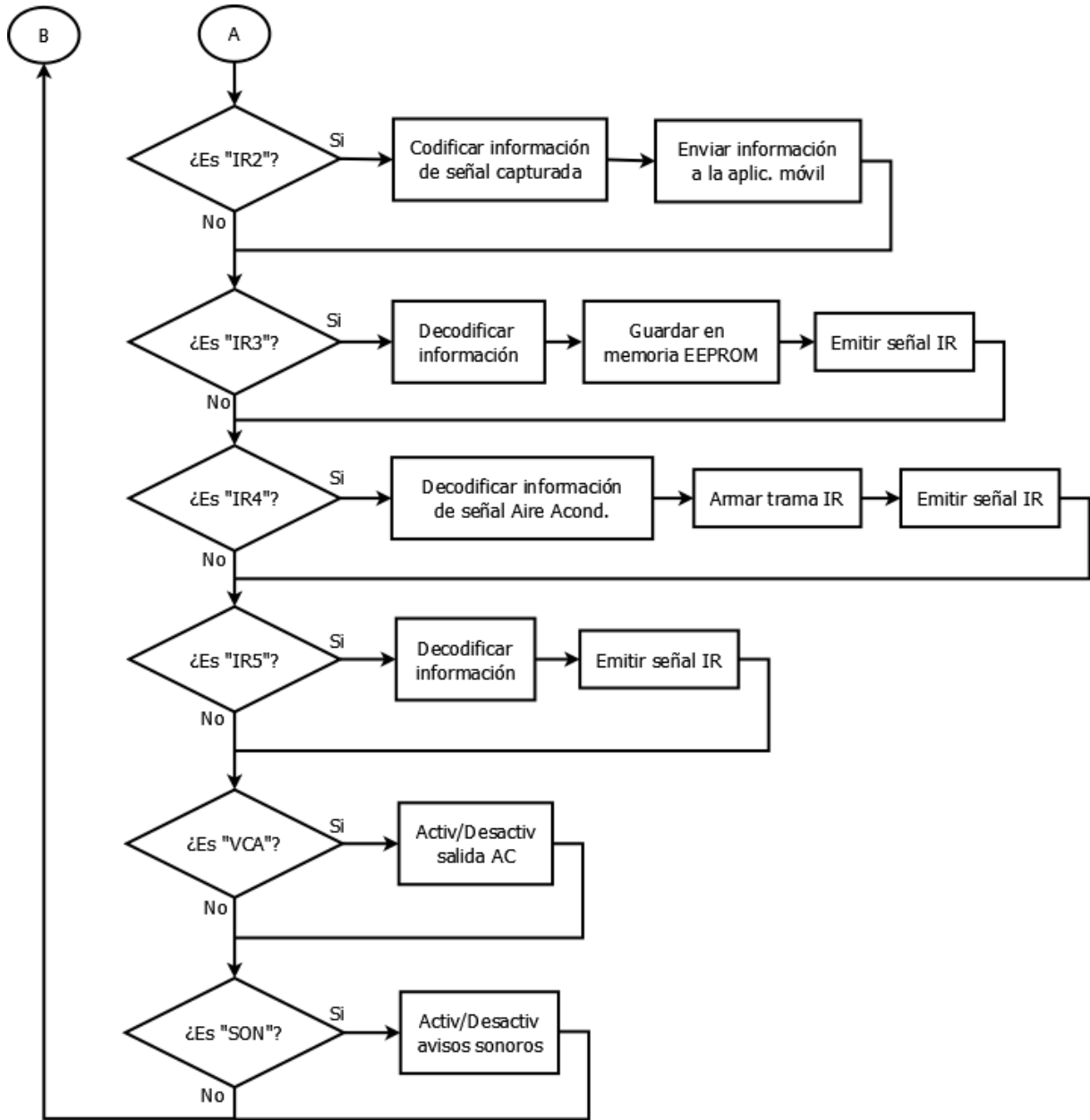


Figura N° 86: Diagrama de flujo de firmware (2).

Capítulo 3: Resultados

Con las placas fabricadas y el firmware finalizado se procedió a realizar las pruebas definitivas en la mayor cantidad de electrodomésticos a los cuales se pudo acceder, se verificó que el sistema pueda recibir y emitir de forma correcta las señales infrarrojas. Se ensayaron 6 televisores LCD, 3 televisores de tubo, 2 reproductores de DVD, 1 reproductor VHS, 2 decodificadores de señal, 5 acondicionadores de aire y 1 equipo de música.

El sistema se desempeñó sin inconvenientes en la mayoría de ellos, dado que en el caso de un televisor Tonomac y otro Samsung, ambos de tubo, el sistema logró decodificar de forma correcta la señal de los controles remotos originales pero en el momento de emitir la misma, los televisores en cuestión no reconocieron la señal. Es posible que la frecuencia de portadora utilizada en este proyecto se encuentre muy alejada de la correcta para estos electrodomésticos. Además, se encontró que un reproductor de DVD Philips utiliza el protocolo RC-6 el cual no está soportado por este sistema por lo que este falla al intentar decodificarlo.

También se observó que en la mayoría de los electrodomésticos no es necesario orientar el emisor infrarrojo hacia los mismos, es decir, la transmisión es omnidireccional. Esto no se logró en los siguientes dispositivos: televisor Ken Brown, reproductor de DVD Tonomac, reproductor VHS Philco, uno de los acondicionadores de aire y un equipo de música Philco. En estos casos es necesario direccionar el diodo emisor, lo cual representa una limitación del sistema.

En relación a la comunicación WiFi se probó utilizar distintos *routers* con el fin de observar si afectaba al desempeño. Se probaron *routers* de la marca Kozumi, Edimax, TP-LINK, en estos casos con la empresa Arnet como proveedor de Internet. También se verificó el funcionamiento con un router CISCO, el cual es provisto a sus clientes por la empresa Fibertel.

En el caso de los primeros *routers* mencionados no se encontró ningún problema, el sistema se conectaba sin dificultad. No obstante, en el caso de CISCO, en algunas ocasiones la conexión fallaba en el primer intento teniendo que volver a indicar que se realice la conexión. En todos los casos, el tiempo de respuesta desde que se presiona un botón en la pantalla del móvil y se observa la acción en el dispositivo en cuestión es prácticamente inmediato.

En cuanto a la aplicación móvil, se verificó el funcionamiento de la misma en diversos dispositivos móviles. En concreto esto se realizó sobre un teléfono Samsung Galaxy Grand Prime con una pantalla de 5" con Android 5.1, un teléfono LG L1 con una pantalla de 3" y Android 4.1 y una *tablet* Samsung Tab 3 con una pantalla de 7" y Android 4.4.

Se observó que la aplicación funciona de manera correcta sobre el teléfono Samsung, lo cual era previsible ya que con este dispositivo se realizó todo el desarrollo del proyecto. Por otro lado, en el teléfono LG, por tener una pantalla de menor tamaño, se observaron algunos controles con un tamaño desproporcionado llegando, en algunos casos, a solaparse u ocultarse entre sí haciendo imposible la correcta utilización. En cuanto a la *tablet*, se observa que la aplicación es completamente funcional, sin embargo, al tener una pantalla relativamente grande, los controles se presentan mal distribuidos o muy separados entre sí lo que genera una apariencia visual no muy agradable.

Capítulo 4: Análisis de costos

A continuación se muestra la lista de componentes y su correspondiente costo que se utilizaron en este proyecto, tal cual fue diseñado. La mayoría de ellos fueron adquiridos en proveedores nacionales, excepto por la placa de desarrollo Stellaris la cual no se comercializa en Argentina pero puede ser comprada directamente al fabricante Texas Instruments, en Estados Unidos.

Cant.	Descripción	Costo
26	Res. 0805	\$2,17
3	Cap. 0805	\$0,79
1	Diodo 4148	\$0,46
4	Triac L0109N	\$30,86
4	Opto. MOC3021M	\$49,88
4	Cap. 100nF/400V	\$7,66
2	Borneras	\$6,70
2	Trans. BC817	\$1,32
1	<i>Buzzer</i>	\$14,61
1	ESP8266	\$150
1	Pulsador	\$2,27
8	Led 0805	\$12,32
1	Reg. TPS76733	\$23,46
1	ULN2803	\$5,85
1	Receptor IR	\$30
1	Led IR	\$2,20
2	Conec. Header Hembra	\$15,94
1	Cap. 10uF/10V Tant.	\$4,01
1	Stellaris LunchPad	\$221
1	PCB	\$344
1	Fuente de alim.	\$80
Total		\$1005,50

Tabla N°2: Costos de componentes del proyecto.

En cuanto a las horas de dedicadas al desarrollo del proyecto, este requirió alrededor de 4 meses y fue ejecutado por solo una persona. En este periodo se llevó a cabo la investigación, el diseño, los ensayos, la documentación, la fabricación de placa y la realización de la aplicación móvil.

Tal como se puede percibir, el valor total de los componentes utilizados sin contabilizar las horas de trabajo es algo elevado. Se puede reducir el costo eliminando algunas prestaciones del sistema, por ejemplo, se podría suprimir el control de cargas AC, se podría quitar el *buzzer* y los *leds* indicadores, también se debería eliminar la placa de desarrollo y sólo utilizar el microcontrolador de la misma. Con estas simplificaciones se reduciría a la mitad el tamaño del PCB por lo que el costo del mismo también sería menor, más aun si se lo fabrica en grandes cantidades y en plazos no tan rígidos.

A continuación se presenta una segunda lista de componentes teniendo en cuenta los comentarios anteriores

Cant.	Descripción	Costo
9	Res. 0805	\$0,75
3	Cap. 0805	\$0,79
1	Trans. BC817	\$0,66
1	ESP8266	\$150
1	Pulsador	\$2,27
1	Reg. TPS76733	\$23,46
1	Receptor IR	\$30
1	Led IR	\$2,20
1	Cap. 10uF/10V Tant.	\$4,01
1	Micro TM4C123GH6PM	\$83,36
1	PCB	\$200 ^(*)
1	Fuente de alim.	\$80
Total		\$577,77

Tabla N°3: Costos de componentes del proyecto reducido.

(*) Estimado.

Se observa que el costo de los componentes se reduce casi en un 50%. Se podría reducir aún más este valor si se considera adquirir estos artículos en cantidades grandes y comprándolos directamente en el exterior, como por ejemplo en las empresas distribuidoras Digikey o Mouser, ubicadas en Estados Unidos.

Capítulo 5: **Discusión y conclusión**

Teniendo en cuenta los resultados las pruebas y ensayos que se realizaron es posible concluir que los objetivos planteados en el inicio de este proyecto han sido logrados de forma aceptable.

Se la logrado conseguir un prototipo que se encuentra muy cercano a una etapa de producción, con un resultado muy profesional tanto en el diseño de las placas, como también en el desarrollo de la aplicación móvil. El dispositivo desarrollado cumple de forma satisfactoria con las necesidades y expectativas del usuario.

Como posibles y necesarias mejoras, que deberían realizarse antes de pasar el diseño a producción, se puede nombrar las siguientes:

1. Se debería mejorar la apariencia visual de la aplicación móvil, esto mejorar lo que comúnmente se conoce como “experiencia de usuario”. Para esto se debería contar con un profesional en este campo, como un diseñador gráfico. Por otro lado, se debería incrementar la compatibilidad entre móviles con diferentes pantallas y resoluciones, esto se lograría implementando múltiples *layouts* de componentes y menús, así la aplicación seleccionaría el adecuado en función del dispositivo en el que se ejecuta. De esta misma forma se podría agregar compatibilidad para distintos idiomas.
2. Tal como las pruebas lo demostraron, no se consiguió que el emisor infrarrojo sea omnidireccional en la totalidad de las situaciones. Esto genera una limitación ya que dependiendo del tipo y de la ubicación de los electrodomésticos en un ambiente, es posible que este quede fuera del campo de transmisión. Para hallar una solución a esto, se deberían realizar pruebas colocando alguna lente o material refractario que distribuya la señal IR en un ángulo mayor.
3. Para que este sistema sea totalmente confiable y profesional es necesario reemplazar los servidores utilizados por otros con mayor seguridad y con la garantía de que no tenga problemas de estabilidad teniendo en cuenta que en un futuro cientos o miles de usuarios subirían y descargarían información del servidor de forma simultánea. Además, se debería contar con un sistema de ranking y denuncias así el usuario puede elegir qué información descargar basado en la puntuación que esta tienen actualmente, y por otro lado si se encuentra que un usuario ha subido datos de un botón en forma incorrecta, con o sin intención, se podría denunciar la situación y un administrador eliminaría del servidor la información errónea.
4. Se deberían incorporar soporte para más protocolos de señales infrarrojas y más fabricantes de equipos acondicionadores de aire. De esta forma se lograría tener mayor compatibilidad y un mercado más grande. Además, se podría discriminar los dispositivos por zonas geográficas ya que no todos los modelos de televisores, por ejemplo, se comercializan en todos los países. Con esta clasificación el usuario, cuando busca en el servidor un botón requerido, lo podría encontrar más rápidamente.
5. Se deberían buscar alternativas para disminuir los costos ya que, como se observa en el capítulo anterior, estos pueden ser considerados como elevados, teniendo en cuenta que no se

incluyeron los costos relacionado a la mano de obre en el desarrollo y tampoco se consideró la carcasa que debería llevar el dispositivo.

Se debe tener presente que este sistema está orientado al uso en el hogar y que no forma parte de una cadena productiva por lo que no genera ningún ingreso para el usuario. Es por esto que los costos deberían reducirse al mínimo posible.

Capítulo 6: Literatura citada

- [1] Santamaría, P. (2014). Domótica y el futuro que siempre está llegando [Mensaje en un blog]. Recuperado de <https://www.xataka.com/domotica-1/domotica-y-el-futuro-que-siempre-esta-llegando>
- [2] Evans, D. (2011). Internet de las cosas: Cómo la próxima evolución de Internet lo cambia todo. Recuperado de http://www.cisco.com/c/dam/global/es_mx/solutions/executive/assets/pdf/internet-of-things-iot-ibsg.pdf
- [3] Grassi, M. (s.f.). Diseño y construcción de un Sistema de Control Remoto Infrarrojo. Recuperado de <http://miguelgrassi.com.ar/arduino/ControlRemotoIR.htm>
- [4] Led. (2017, 7 de julio). En Wikipedia, la enciclopedia libre. Recuperado el 7 de julio de 2017 a las 21:30 de <https://es.wikipedia.org/w/index.php?title=Led&oldid=100315138>
- [5] Bergmans, S. (2002). IR Remote Control [Mensaje en un blog]. Recuperado de <http://www.sbprojects.com/knowledge/ir/index.php>
- [6] Wifi. (2017, 5 de julio). En Wikipedia, la enciclopedia libre. Recuperado el 7 de julio de 2017 a las 21:34 de <https://es.wikipedia.org/w/index.php?title=Wifi&oldid=100285890>
- [7] Hypertext Transfer Protocol. (2017, 12 de junio). En Wikipedia, la enciclopedia libre. Recuperado el 7 de julio de 2017 a las 21:35 de https://es.wikipedia.org/w/index.php?title=Hypertext_Transfer_Protocol&oldid=99796071
- [8] Sitio oficial de Android. Visible en <https://developer.android.com/about/android.html>
- [9] Sierra Gracia, M. (2006). ¿Qué es un servidor y cuáles son los principales tipos de servidores? Recuperado de <http://www.aprenderaprogramar.com/attachments/article/542/DV00408A%20Que%20es%20un%20servidor%20principales%20tipos%20proxy%20smtp%20ftp%20web%20dns.pdf>
- [10] Introducción a JSON. (s.f.). Recuperado de <http://www.json.org/json-es.html>
- [11] IR Remote Control. Disponible en Play Store de Google.
- [12] Asmart Remote IR. Disponible en Play Store de Google.
- [13] Universal TV Remote. Disponible en Play Store de Google.
- [14] Microchip. Hoja de datos PIC18F14K50. Recuperado de <http://www.microchip.com/wwwproducts/en/PIC18F14K50>
- [15] Texas Instruments. Stellaris LunchPad. Recuperado de <http://www.ti.com/tool/ek-lm4f120xl>
- [16] Prometec. (s.f.). Arduino y Wifi ESP8266 [Mensaje en un blog]. Recuperado de <http://www.prometec.net/arduino-wifi/>
- [17] Pridopia. (s.f.). ESP8266 AT Command Set. Recuperado de <http://www.pridopia.co.uk/pi-doc/ESP8266ATCommandsSet.pdf>
- [18] Microchip. RN171. Recuperado de <http://www.microchip.com/wwwproducts/en/RN171>
- [19] García, L. (s.f.). Un poco de JSON [Mensaje en un blog]. Recuperado de <https://unpocodejava.wordpress.com/2010/02/18/un-poco-de-json/>
- [20] Motorola. Batería BC50. Recuperado de http://www.test.cellphonepartscanada.com/mobile_detail_acc.asp?pid=MOTOBATBC